Maestría en

CIBERSEGURIDAD

Trabajo previo a la obtención de título de Magíster en Ciberseguridad

AUTOR/ES:

Blacio Edmundo Velasco Miranda

Diego Gabriel Villegas Vásconez

Santiago Fernando Sánchez Viteri

Carlos José Calderón Cabezas

TUTOR/ES:

Iván Galo Reyes Chacón

Alejandro Cortés López

Análisis Forense Digital de un SIEM no WAZUH

Diciembre - 2025

Quito - Ecuador

REINVENTEMOS EL FUTURO

Certificación de Autoría

Nosotros, **Blacio Edmundo Velasco Miranda, Diego Gabriel Villegas Vásconez, Santiago Fernando Sánchez Viteri y Carlos José Calderón Cabezas**, declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido presentado anteriormente para ningún grado o calificación profesional y que se ha consultado la bibliografía detallada.

Cedemos nuestros derechos de propiedad intelectual a la Universidad Internacional del Ecuador (UIDE), para que sea publicado y divulgado en internet, según lo establecido en la Ley de Propiedad Intelectual, su reglamento y demás disposiciones legales.

Firma
Blacio Edmundo Velasco Miranda

Firma
Diego Gabriel Villegas Vásconez

Firma
Santiago Fernando Sánchez Viteri

Firma
Carlos José Calderón Cabezas

Autorización de Derechos de Propiedad Intelectual

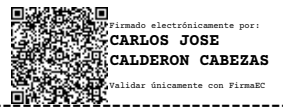
Nosotros, **Blacio Edmundo Velasco Miranda, Diego Gabriel Villegas Vásconez, Santiago Fernando Sánchez Viteri y Carlos José Calderón Cabezas**, en calidad de autores del trabajo de investigación titulado ***Análisis Forense Digital de un SIEM no WAZUH***, autorizamos a la Universidad Internacional del Ecuador (UIDE) para hacer uso de todos los contenidos que nos pertenecen o de parte de los que contiene esta obra, con fines estrictamente académicos o de investigación. Los derechos que como autores nos corresponden, lo establecido en los artículos 5, 6, 8, 19 y demás pertinentes de la Ley de Propiedad Intelectual y su Reglamento en Ecuador.

D. M. Quito, 9 de enero de 2026

Firma
Blacio Edmundo Velasco Miranda

Firma
Diego Gabriel Villegas Vásconez

Firma
Santiago Fernando Sánchez Viteri



Firma
Carlos José Calderón Cabezas

Aprobación de Dirección y Coordinación del Programa

Nosotros, **Iván Galo Reyes Chacón y Alejandro Cortés López,** declaramos que: **Blacio Edmundo Velasco Miranda, Diego Gabriel Villegas Vásconez, Santiago Fernando Sánchez Viteri y Carlos José Calderón Cabezas** son los autores exclusivos de la presente investigación y que ésta es original, auténtica y personal de ellos.

Alejandro Cortés L.

Maestría en Ciberseguridad

Iván Reyes Ch.

Maestría en Ciberseguridad

Dedicatoria

Dedicamos este trabajo a nuestras familias, quienes han sido el pilar fundamental en cada paso de nuestro camino académico.

Gracias por su amor incondicional, su paciencia infinita durante nuestras largas noches de estudio y por creer en nosotros incluso cuando el camino se tornaba difícil.

A todas aquellas personas que, con una palabra de aliento o un gesto de apoyo, nos impulsaron a no rendirnos y a culminar con éxito esta importante meta.

Este logro es tan nuestro como de ustedes.

Agradecimientos

Deseamos expresar nuestro más sincero agradecimiento a la Universidad Internacional del Ecuador y a la Escuela Internacional de Gerencia, por abrirnos las puertas al conocimiento y permitirnos formar parte de sus prestigiosos programas de Maestrías en especial la Maestría en Ciberseguridad.

Gracias por brindarnos las herramientas y el entorno académico necesario para nuestro crecimiento profesional.

Extendemos un agradecimiento especial a nuestros tutores, Ing. Ivan Galo Reyes Chacón, Msc e Ing. Alejandro Cortés López, Msc, por su invaluable guía, paciencia y transferencia de conocimientos.

Sus orientaciones técnicas y metodológicas fueron decisivas para la correcta ejecución y culminación de este proyecto de análisis forense.

Del mismo modo, agradecemos a todos los integrantes del Grupo 7: Blacio Velasco, Carlos Calderón, Diego Villegas y Santiago Sánchez, por el compañerismo, el trabajo en equipo y la sinergia lograda.

La colaboración y el intercambio de experiencias entre nosotros fueron la clave para alcanzar los objetivos planteados en este Proyecto PBL.

Resumen

El presente proyecto de la Maestría en Ciberseguridad aborda el Análisis Forense Digital de un sistema SIEM basado en la pila ELK Stack (Elasticsearch, Logstash y Kibana), desplegado sobre un entorno controlado de virtualización.

El objetivo principal fue reconstruir la línea de tiempo de un ciberataque, identificar los vectores de entrada y determinar el alcance del compromiso en una infraestructura de monitoreo centralizada.

Para la ejecución, se implementó una topología de red utilizando GNS3 y VMware, integrando un servidor Ubuntu (SIEM), un cliente Windows 10 y estaciones de ataque y análisis con Kali Linux.

La metodología se dividió en dos fases críticas: la ofensiva, donde se ejecutaron simulaciones de ataques de fuerza bruta (Hydra), escaneo de vulnerabilidades (Nmap), creación de usuarios ilegítimos ("atacanteg7") e inyección de payloads mediante Metasploit; y la fase defensiva/forense, enfocada en la adquisición y análisis de evidencias.

El proceso forense incluyó la extracción de memoria RAM en caliente utilizando LiME y la creación de imágenes de disco forenses vía red mediante dcfldd, asegurando la integridad de los datos mediante hashing.

El análisis posterior, realizado con Autopsy 4.22 y Volatility3, permitió identificar artefactos críticos: manipulación de logs, credenciales expuestas en texto claro, sesiones SSH no autorizadas y procesos anómalos (como psimon e inyecciones en procesos Java) vinculados a la actividad del atacante.

Los resultados confirman la capacidad de detectar intrusiones y reconstruir eventos de seguridad mediante el análisis correlacionado de logs y memoria en entornos SIEM no propietarios.

Palabras Clave: Ciberseguridad, Análisis Forense Digital, SIEM, ELK Stack, Volatility3, Autopsy, Respuesta a Incidentes.

Abstract

This project, part of the Master in Cybersecurity, addresses the Digital Forensic Analysis of a SIEM system based on the ELK Stack (Elasticsearch, Logstash, and Kibana), deployed within a controlled virtualization environment.

The main objective was to reconstruct the timeline of a cyberattack, identify entry vectors, and determine the scope of the compromise within a centralized monitoring infrastructure.

For the execution, a network topology was implemented using GNS3 and VMware, integrating an Ubuntu server (SIEM), a Windows 10 client, and attack/analysis stations running Kali Linux.

The methodology was divided into two critical phases: the offensive phase, which involved simulated brute-force attacks (Hydra), vulnerability scanning (Nmap), creation of illegitimate users ("atacanteg7"), and payload injection using Metasploit; and the defensive/forensic phase, focused on evidence acquisition and analysis.

The forensic process included live RAM extraction using LiME and forensic disk imaging via network using dcfldd, ensuring data integrity through hashing.

Subsequent analysis, performed with Autopsy 4.22 and Volatility3, allowed for the identification of critical artifacts: log manipulation, cleartext credentials exposure, unauthorized SSH sessions, and anomalous processes (such as psimon and injections into Java processes) linked to the attacker's activity.

The results confirm the ability to detect intrusions and reconstruct security events through the correlated analysis of logs and memory in non-proprietary SIEM environments.

Keywords: Cybersecurity, Digital Forensics, SIEM, ELK Stack, Volatility3, Autopsy, Incident Response.

Índice de Contenidos

Índice de Tablas

Índice de Figuras

Anexos

1. Introducción

1.1. Definición del proyecto

El presente proyecto consiste en la realización de un Análisis Forense Digital a un sistema SIEM (Security Information and Event Management) que no utiliza la plataforma Wazuh, optando en su lugar por la implementación de la pila ELK Stack (Elasticsearch, Logstash y Kibana). El proyecto se centra en investigar incidentes de seguridad mediante el uso de los registros y datos recopilados por este sistema centralizado.

Para su ejecución, se diseñó y montó un laboratorio controlado de virtualización que simula una infraestructura de red real. Este entorno incluye un servidor Ubuntu configurado como SIEM con ElasticStack 8.19.7, una máquina víctima con Windows 10, y equipos con Kali Linux desempeñando roles tanto de atacante (para generar incidentes reales) como de analista forense. El núcleo del proyecto es validar la capacidad de rastrear, preservar y analizar evidencia digital dentro de esta arquitectura específica.

1.2. Justificación e importancia del trabajo de investigación

La implementación de un equipo de respuesta a incidentes eficaz requiere procedimientos sólidos para identificar, recuperar y analizar evidencias, tareas propias de la Ciencia Forense Digital. La importancia de esta investigación radica en abordar el análisis forense en un entorno SIEM "no Wazuh" (específicamente ELK Stack). A diferencia de plataformas como Wazuh que ya integran capacidades de EDR y análisis de comportamiento, un SIEM puro basado en ELK requiere un enfoque forense especializado

para extraer y correlacionar datos de fuentes diversas como firewalls, servidores y endpoints.

Este trabajo es crucial para demostrar cómo, mediante el uso de herramientas como Elasticsearch, Logstash y Kibana, es posible reconstruir la línea de tiempo de un ataque, identificar el punto de entrada y determinar el alcance de una brecha de seguridad. Además, el proyecto aporta valor al documentar metodologías para generar informes legalmente admisibles y mejorar las estrategias de defensa de las organizaciones ante incidentes de ciberseguridad.

1.3. Alcance

El alcance de este proyecto abarca las siguientes áreas y actividades limitadas al entorno de laboratorio desplegado:

Infraestructura: Implementación de un laboratorio virtual utilizando GNS3 y VMware Workstation Pro, compuesto por un enrutador, un servidor Ubuntu 24.04 (SIEM), un cliente Windows 10 y estaciones Kali Linux.

Despliegue del SIEM: Instalación y configuración completa del Elastic Stack (Java, Elasticsearch, Logstash, Kibana y Filebeat) para la recolección y visualización de logs.

Simulación de Amenazas: Ejecución controlada de ciberataques contra la infraestructura del SIEM, incluyendo escaneos de red (Nmap), ataques de fuerza bruta

(Hydra), creación de usuarios no autorizados y uso de frameworks de explotación (Metasploit) para generar "ruido" y registros de eventos.

Adquisición Forense: Extracción de evidencia digital volátil (memoria RAM) utilizando LiME en tiempo real y adquisición de imágenes de disco forenses mediante la herramienta dcfldd, garantizando la integridad mediante hash (MD5/SHA256).

Análisis Forense: Examen detallado de los artefactos recolectados utilizando herramientas especializadas como Autopsy 4.22.1 (para análisis de disco y logs) y Volatility3 (para análisis de memoria), con el fin de detectar procesos sospechosos, conexiones anómalas y manipulación del sistema.

1.4. Objetivos

1.4.1. Objetivo General

Realizar el análisis forense digital de un sistema SIEM basado en ELK Stack (no Wazuh) para reconstruir la línea de tiempo de un ataque informático, identificando el vector de entrada, el alcance del compromiso y la atribución de las acciones maliciosas, con el fin de mejorar las defensas de la infraestructura.

1.4.2. Objetivos Específicos

- Implementar un entorno de laboratorio virtualizado con la pila ELK (Elasticsearch, Logstash, Kibana) sobre Ubuntu Server para centralizar y monitorear los eventos de seguridad de la red.

- Ejecutar simulaciones de ataques cibernéticos (fuerza bruta, escaneo de puertos, inyección de payloads) contra el servidor SIEM para generar registros de incidentes y evidencias digitales trazables.

- Adquirir evidencias digitales volátiles y no volátiles del sistema comprometido utilizando metodologías forenses y herramientas como LiME y dcfldd, asegurando su preservación mediante cálculo de hashes.

- Analizar las evidencias recolectadas (imágenes de disco y volcados de memoria) mediante software forense (Autopsy y Volatility) para identificar usuarios ilegítimos, procesos maliciosos y configuraciones vulneradas.

- Documentar y presentar los hallazgos del análisis para validar la secuencia del ataque y la efectividad del monitoreo del SIEM ante los incidentes generados.

2. Revisión de Literatura

2.1. Estado del Arte

En los últimos años, la investigación sobre análisis forense digital ha enfatizado el uso de plataformas SIEM alternativas a Wazuh, como ELK Stack (Elastic Security), Splunk o Graylog, para centralizar y correlacionar grandes volúmenes de registros de seguridad.

Estos sistemas SIEM recopilan eventos de red, sistemas y aplicaciones, aplican reglas de correlación y análisis avanzado (incluyendo aprendizaje automático) para detectar amenazas y anomalías. Bhardwaj et al. (2025) ejemplifican este enfoque: implementaron un SIEM personalizado basado en Elasticsearch que correlaciona logs de Sysmon para detectar ataques de escalada de privilegios, confirmando que eventos específicos de registro (IDs 12 y 7) brindan “evidencia forense crítica” al reconstruir la cadena de ataque.

De igual forma, se han propuesto marcos proactivos que integran SIEM con marcos de amenazas y normas internacionales. Yang et al. (2025) presentan un procedimiento operativo estándar (P-DEFSOP) que combina SIEM/MDR con el marco MITRE ATT&CK y herramientas forenses automatizadas, mejorando la detección y el análisis oportuno de amenazas mientras se garantiza la integridad de la evidencia digital. Además, advierten que los SIEM convencionales suelen pasar por alto una gran proporción de técnicas ATT&CK: se estima que detectan apenas el 21% de los métodos conocidos, lo que motiva la integración de inteligencia de amenazas proactiva. Otra línea de investigación reciente es la reconstrucción de cronologías de ataque mediante grafos de procedencia: por ejemplo, SecTracer (Lee et al., 2025) emplea redes definidas por software (SDN) para recolectar

datos y construir grafos causales que permiten visualizar de manera clara la historia completa de una intrusión.

En cuanto a herramientas concretas, la literatura compara la eficacia de distintas soluciones SIEM en el contexto forense. Sajan et al. (2025) describen herramientas como Graylog –plataforma de código abierto con paneles personalizables para gestión centralizada de logs–, Splunk –solución comercial muy usada con capacidades avanzadas de indexación y visualización– y ELK Stack –suite open source (Elasticsearch, Logstash, Kibana) escalable para ingesta y análisis de logs.

Estudios recientes demuestran que cada plataforma tiene ventajas y limitaciones: por ejemplo, una tesis comparativa de 2025 encontró que SIEM básicos (como Syslog-ng) detectan incidentes más rápido, mientras que soluciones avanzadas como Splunk ofrecen análisis más consistentes. Sin embargo, la tendencia general es aprovechar la capacidad de análisis de logs de estas herramientas para impulsar la respuesta a incidentes y la investigación forense. Asimismo, se observa una evolución hacia la incorporación de técnicas de aprendizaje automático y análisis de comportamiento en SIEM, con el fin de reducir falsos positivos y adaptarse a tácticas de ataque emergentes.

Actualmente, se muestra un enfoque cada vez más integrado: los entornos SIEM modernos combinan fuentes de datos heterogéneas, correlación en tiempo real y marcos de amenazas actualizados (p. ej. MITRE ATT&CK) para facilitar la extracción de evidencias, la reconstrucción del ataque y la respuesta rápida ante incidentes.

2.2. Marco Teórico

El análisis forense digital se define como la disciplina encargada de detectar, adquirir, procesar y analizar datos almacenados en medios electrónicos. Su objetivo principal es extraer de esos datos la información necesaria para reconstruir eventos delictivos y presentar conclusiones como evidencia admisible en procesos legales. En la práctica, esto implica seguir procesos metodológicos que preserven la integridad de la evidencia (cadena de custodia, cifrado, hashing) y aplicar técnicas de correlación y cronología para reconstruir incidentes. Por analogía con la investigación criminal, cada registro de evento (log) actúa como un “punto de datos” que ayuda a los peritos a ver la secuencia de eventos ocurridos en el sistema. En este sentido, los registros de eventos son la fuente primaria de evidencia digital; representan el “repositorio” de información sobre la actividad delictiva y deben recolectarse antes de reiniciar o modificar los sistemas afectados. Graylog señala que es esencial “procesar y analizar” estos datos de logs para integrarlos en la investigación, alimentando los sistemas SIEM y SOAR y previniendo incidentes futuros.

Por otro lado, las plataformas SIEM (Security Information and Event Management) centralizan los logs de múltiples fuentes –red, endpoints, servidores, aplicaciones, nube, dispositivos de seguridad, etc.– y aplican reglas de correlación para detectar patrones sospechosos. Un SIEM permite consolidar datos de seguridad y analizarlos desde un único punto de vista, lo que minimiza el riesgo de pasar por alto incidentes críticos.

Los SIEM modernos suelen incorporar aprendizaje automático para ajustar automáticamente la precisión de las alertas y detectan desviaciones del comportamiento normal. Además, es común mapear los eventos detectados a tácticas y técnicas conocidas

(MITRE ATT&CK), enriqueciendo así el análisis forense con inteligencia de amenazas. Las fases de respuesta a incidentes normalizadas (por ejemplo, ISO/IEC 27035) incluyen una etapa de investigación forense, donde se emplean estos registros para identificar la causa raíz, el origen y el alcance del incidente.

Existen múltiples herramientas que implementan estas funciones. Dentro de las alternativas a Wazuh, se destacan ELK Stack, Splunk y Graylog. ELK (Elastic Stack) es una suite de código abierto que permite ingerir, almacenar, buscar y visualizar grandes volúmenes de logs de forma escalable. Splunk Enterprise es una solución comercial de alto rendimiento que indexa y analiza datos generados por máquinas, ofreciendo avanzadas interfaces de visualización y alertas. Graylog, por su parte, es una plataforma open source para gestión centralizada de registros, que soporta paneles y búsquedas personalizables. Teóricamente, todas estas plataformas permiten crear vistas cronológicas de eventos, generar alertas en base a reglas o ML, y colaborar en la investigación de incidentes.

En conjunto, las herramientas SIEM proporcionan la infraestructura para la recolección forense de evidencias digitales, facilitando la extracción de artefactos (archivos de log, metadatos, tráfico de red) y su análisis en el contexto de la reconstrucción de ataques.

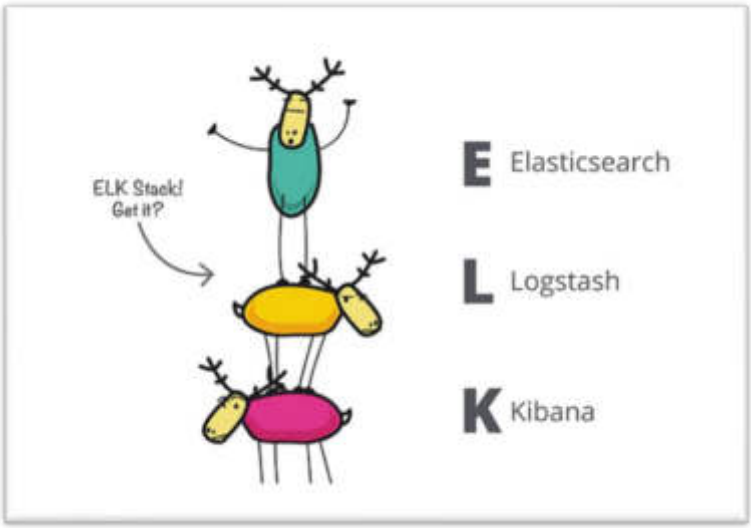
3. Desarrollo

Como parte del análisis y estudio de la Tecnología SIEM NO WAZUH, el grupo de trabajo se ha propuesto realizar el análisis al sistema SIEM con ELK STACK (E = Elasticsearch, L = Logstash, K = Kibana, Pila).

ELK STACK

Figura 1

ELK STACK



Nota: Fuente: https://www.hiberus.com/crecemos-contigo/que-es-un-stack-de-elk-y-que-ventajas-tiene-implementarlo/

¿Qué es un STACK de ELK?

Un stack de ELK es una combinación de tres herramientas de código abierto: Elasticsearch, Logstash y Kibana, que trabajan juntas para proporcionar una solución completa de análisis y monitoreo de registros.

Descripción del Entorno de Pruebas

Para el presente análisis forense se implementará un laboratorio controlado conformado por cuatro equipos virtuales, cada uno con un rol específico dentro del entorno:

Equipo Router: Servirá como enrutador principal, permitiendo la salida hacia la nube y facilitando la conectividad entre los distintos componentes del laboratorio.

Equipo con Ubuntu: En este servidor se instalará y configurará la suite ELK Stack (Elasticsearch, Logstash y Kibana), la cual permitirá la recolección, gestión y visualización de los registros generados durante las pruebas.

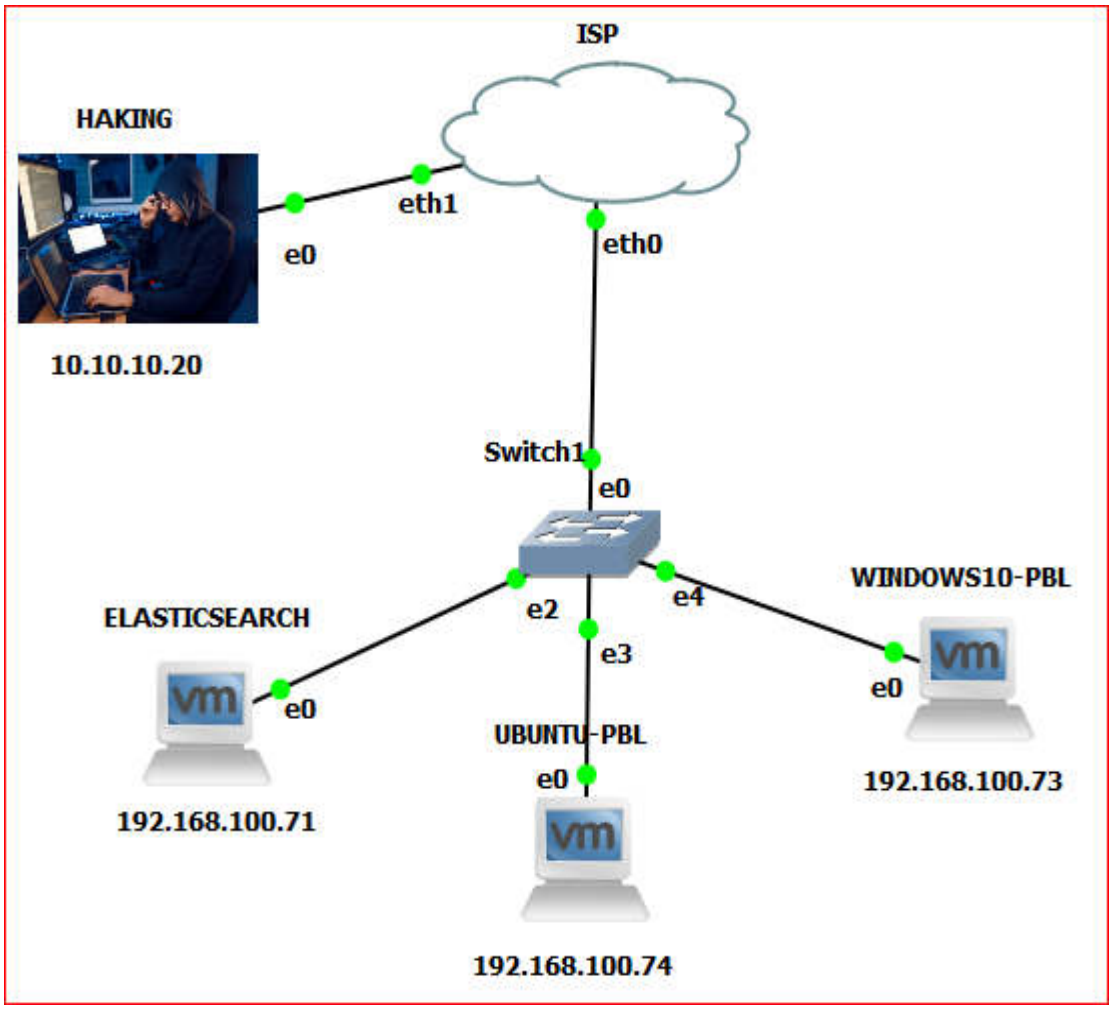
Equipo con Kali Linux (Atacante): Este dispositivo virtual representará el rol de cracker. Será responsable de ejecutar ataques dirigidos contra la infraestructura del laboratorio, intentando comprometerla e introducir malware para simular incidentes de seguridad reales.

Equipo con Kali Linux (Analista Forense): Este equipo se encargará de la monitorización en tiempo real de las actividades registradas en el servidor con ELK Stack. Además, realizará la extracción de la memoria RAM del sistema comprometido con el objetivo de efectuar un análisis forense detallado.

Topología

Figura 2

Topología



Primero descargamos UBUNTU 24.0.4 (ubuntu-24.04.3-live-server-amd64.iso) para luego instalar en la máquina virtual VMware Workstation Pro del siguiente enlace:

`https://ubuntu.com/download/server`

Una vez descargada procedemos a instalar el iso en nuestro VMware, la clave y el usuario por defecto que se colocaron son elk, escogemos la opción de crear nueva máquina virtual, dejamos por defecto tipìcal, escogemos la ruta donde está el iso.

Figura 3

Instalación de Workstation Pro

Le ponemos un nombre a la imagen que se va a crear y escogemos la ruta donde se va a crear, el espacio en disco lo dejamos en 100GB, y por defecto la segunda opción para crear el particionado en el disco.

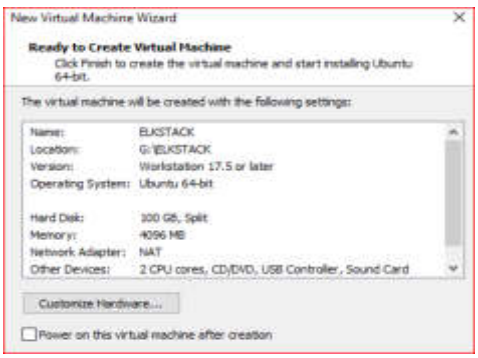
Figura 4

Parámetros de instalación

Nos da la descripción del disco creado y le damos a finalizar.

Figura 5

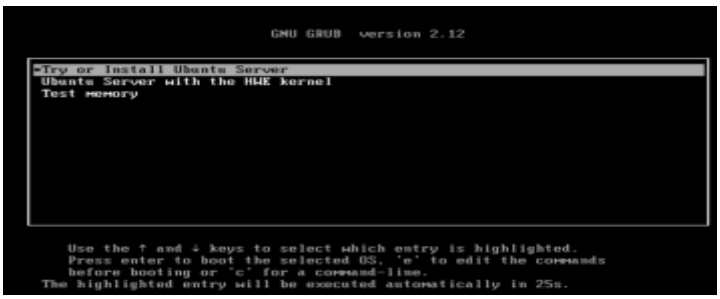
Confirmación de parámetros



Nos lanza a la interfaz de instalación, escogemos la primera opción

Figura 6

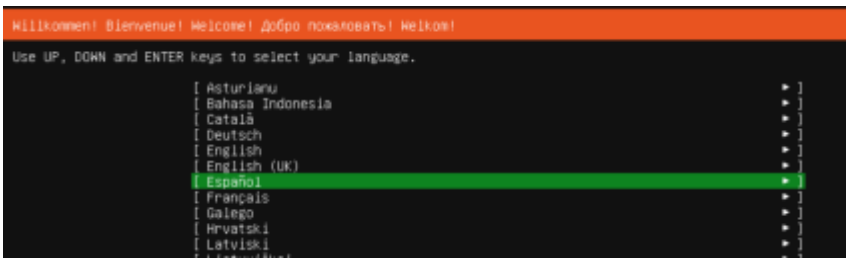
Arranque inicial de la Máquina Virtual (VM)



Escogemos el lenguaje en español

Figura 7

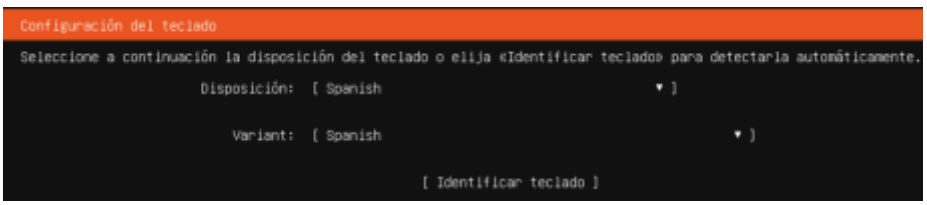
Selección de idioma de la VM



Configuramos el teclado en español y damos enter en HECHO

Figura 8

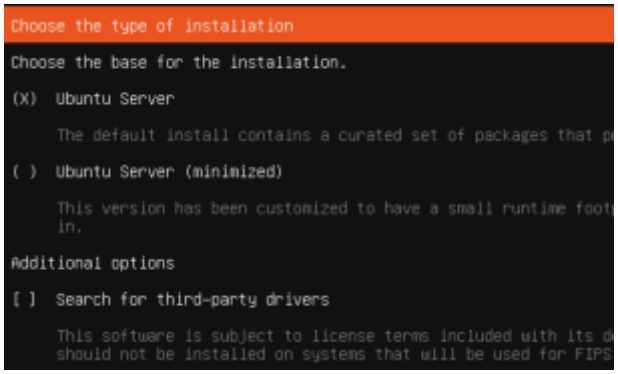
Configuración del teclado en español de VM



Escogemos el tipo de instalación, en nuestro caso escogemos la primera opción de servidor y damos enter en HECHO

Figura 9

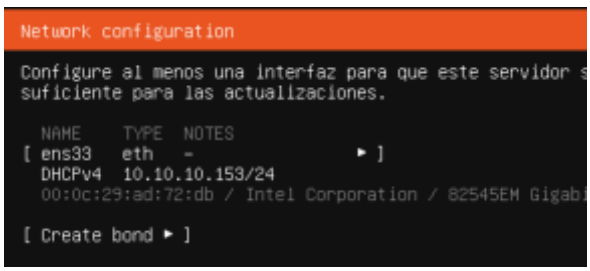
Tipo de instalación de VM



Nos pide configurar la red, en nuestro caso la dejamos por DHCP para poder trabajar remotamente que es el objetivo de este tema de análisis y damos en HECHO,

Figura 10

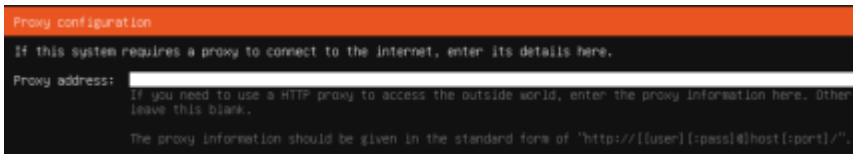
Configuración de red de VM



Nos va a pedir el proxy en este caso por defecto y damos en HECHO.

Figura 11

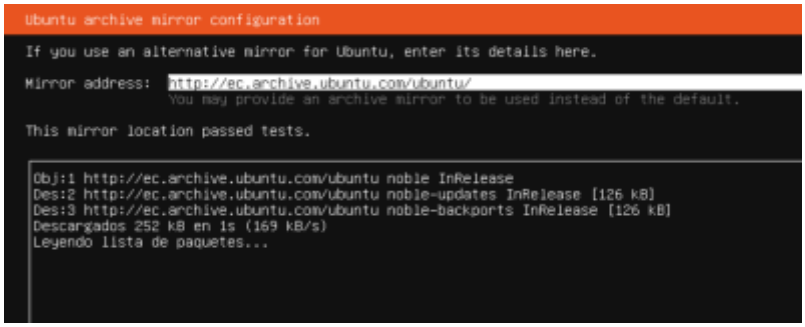
Configuración de proxy de VM



Se conecta al servidor de Ubuntu y va a comenzar a descargarse las actualizaciones para luego instalarlas, damos en HECHO

Figura 12

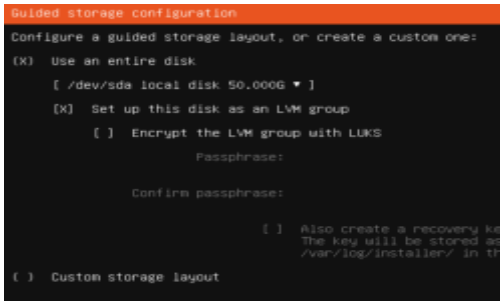
Configuración de actualizaciones de Ubuntu



Nos pide que escojamos donde se va a ejecutar el ISO en este caso colocamos en el disco de 100GB que creamos y damos en HECHO

Figura 13

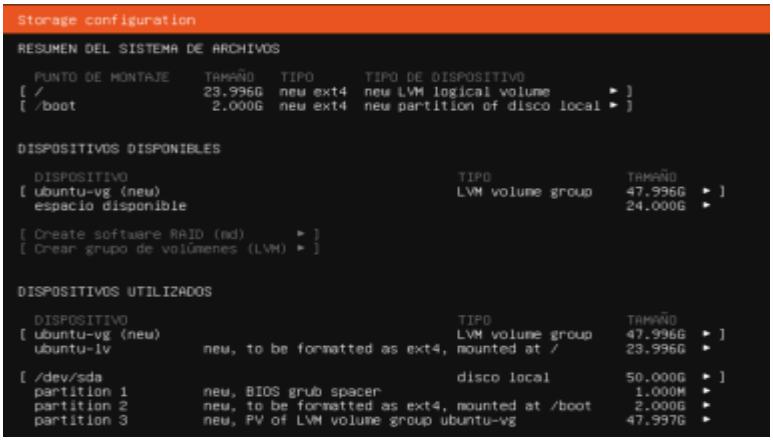
Selección de ISO



Nos da el resumen del sistema de archivos que va a crear a lo cual aceptamos y damos en HECHO

Figura 14

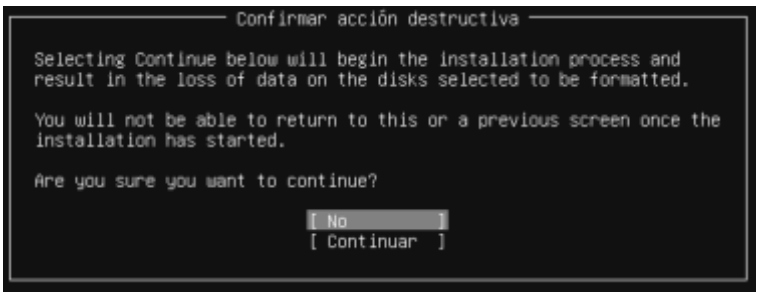
Configuración del almacenamiento



Confirmamos la instalación y damos en continuar

Figura 15

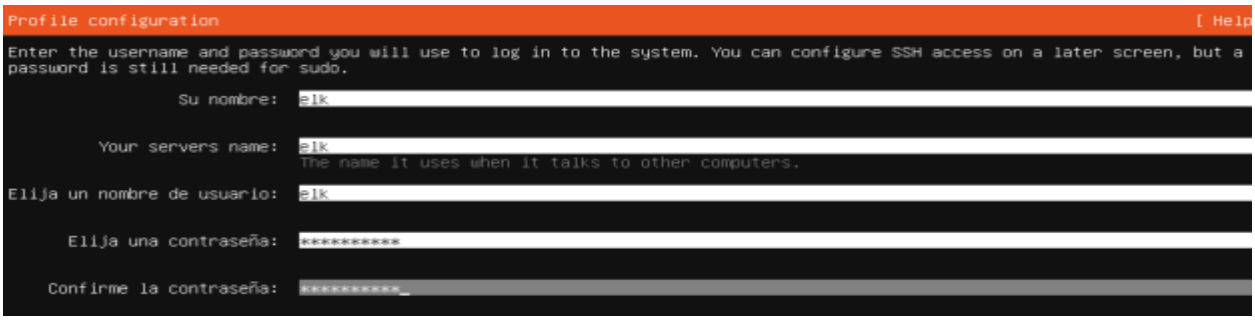
Confirmación de la instalación



Nos pide que llenemos los siguientes datos: nombre de la máquina (elk), nombre del servidor (elk), nombre de usuario (elk) y una contraseña la misma que será para todas las configuraciones en este equipo (elkstackG7)

Figura 16

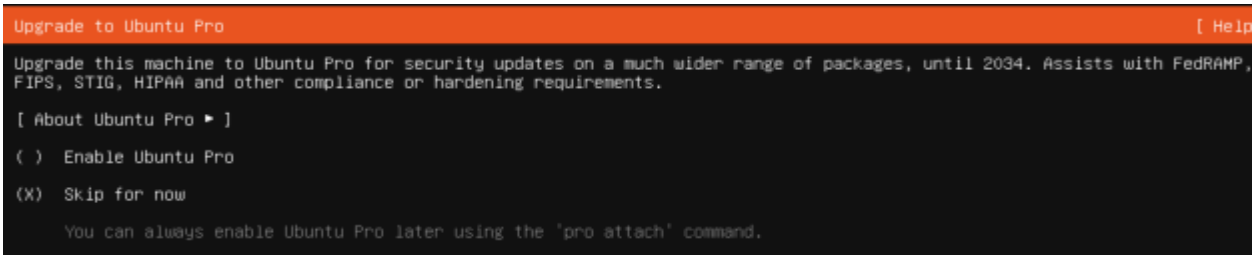
Configuración de usuario



Damos en HECHO y nos pedirá una confirmación, le dejamos por defecto y damos en CONTINUAR

Figura 17

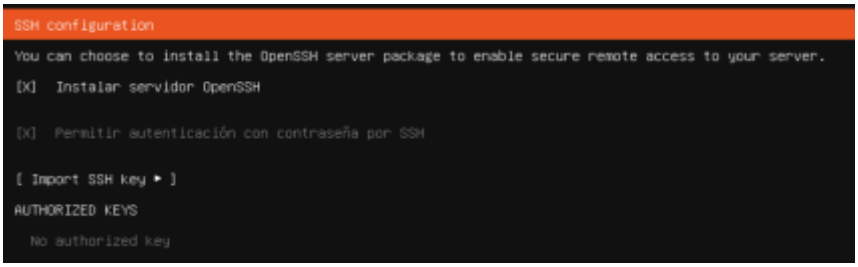
Ventana de confirmación



Marcamos e Instalamos el servidor SSH y damos en HECHO

Figura 18

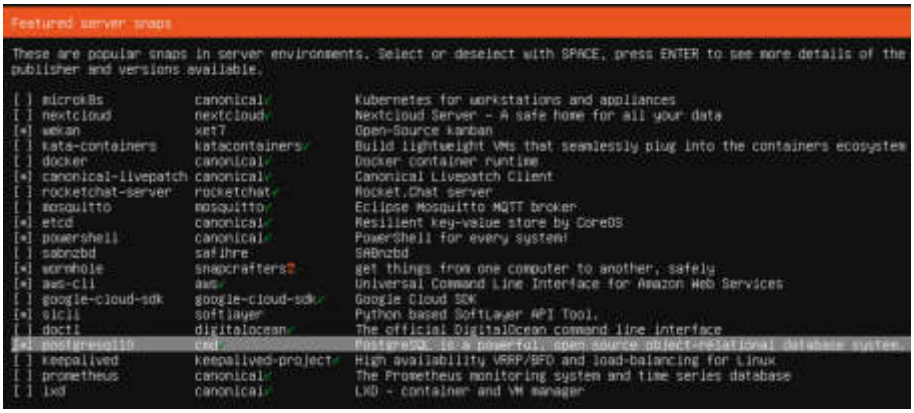
Configuración de SSH



Nos pide escoger que otros servicios deseamos instalar, en nuestro caso ninguna y damos ha HECHO

Figura 19

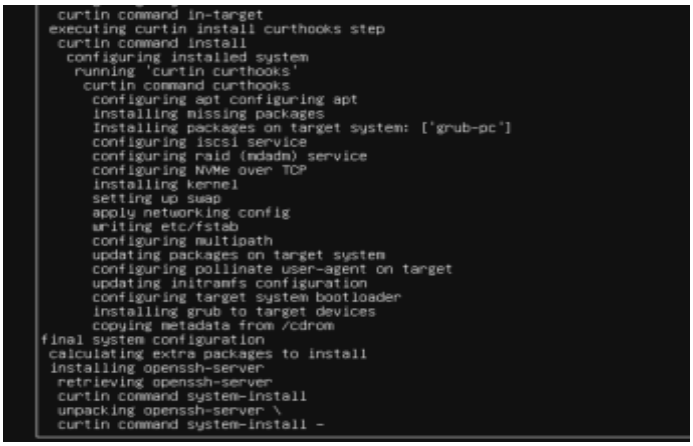
Configuración de servicios para instalar



Y comienza la instalación

Figura 20

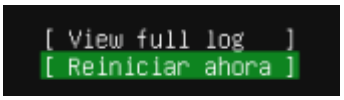
Inicio de la instalación



Le damos clic a reiniciar

Figura 21

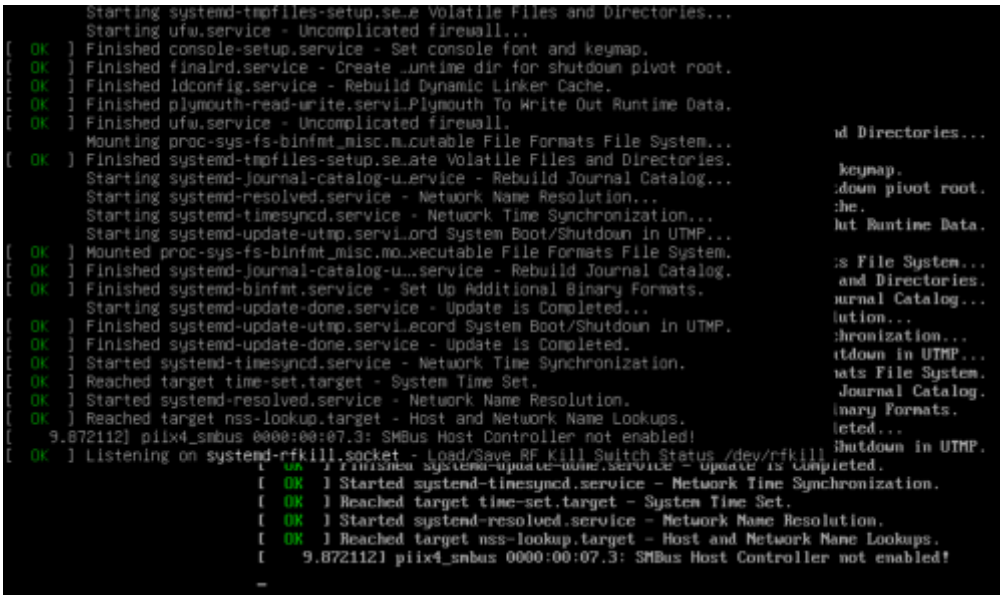
Confirmación de reinicio



Y esperamos a que comience la carga de la imagen

Figura 22

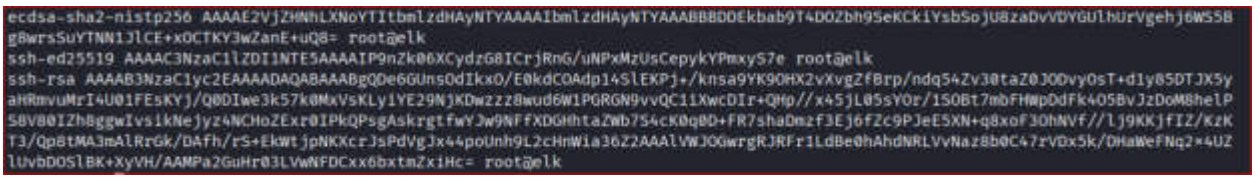
Carga de imagen



Para veracidad de la imagen nos imprime los SHA256 y las claves del sistema

Figura 23

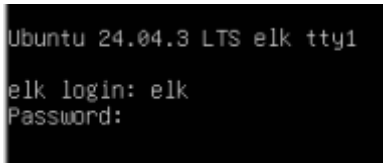
Su SHA256 inicial



Una vez culminada la instalación no pedirá que ingresemos con nuestro usuario y nuestra clave

Figura 24

Login inicial en VM



Si todo es correcto ingresaremos al sistema donde procedemos a configurar el SSH

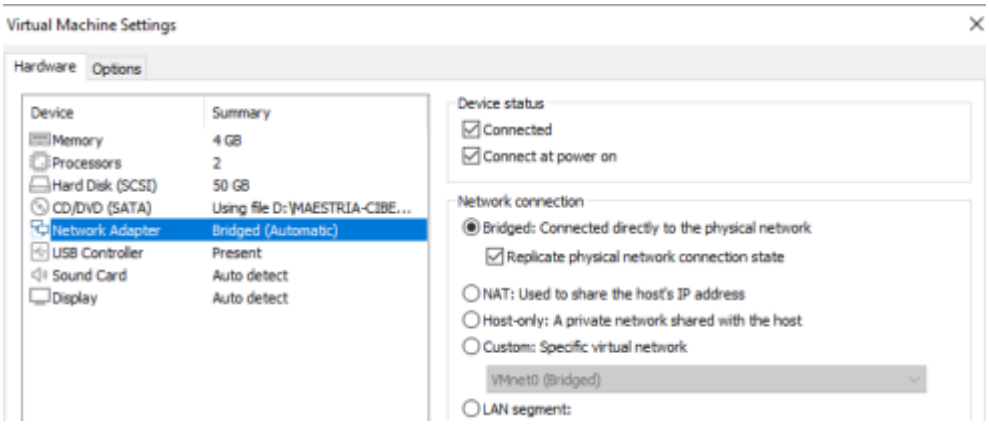
Figura 25

Configuración posterior de SSH

A la máquina virtual la configuramos como Bridged para poder instalar las actualizaciones y el ELK-STACK

Figura 26

Selección de adaptador de red de tipo “puente” en VM



Una vez ejecutado y corriendo el server procedemos a realizar las diferentes actualizaciones, instalaciones y configuraciones.

Instalar Java

Dado que Elasticsearch está desarrollado en Java, es necesario instalar Java. Antes de eso, actualizamos el sistema ejecutando los siguientes comandos:

```
sudo apt update
sudo apt upgrade -y
```

Instalaremos OpenJDK 17, la última versión estable LTS que proporciona Java.

Procedemos a instalarla desde los repositorios estándar de Ubuntu mediante el gestor de paquetes APT.

```
sudo apt install openjdk-17-jdk -y
```

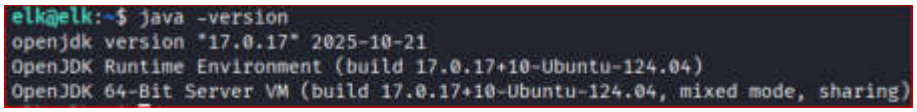
Figura 27

Instalación de OpenJDK

Para confirmar que Java está instalado, ejecute el comando: `java -versión`

Figura 28

Comprobación de instalación de Java



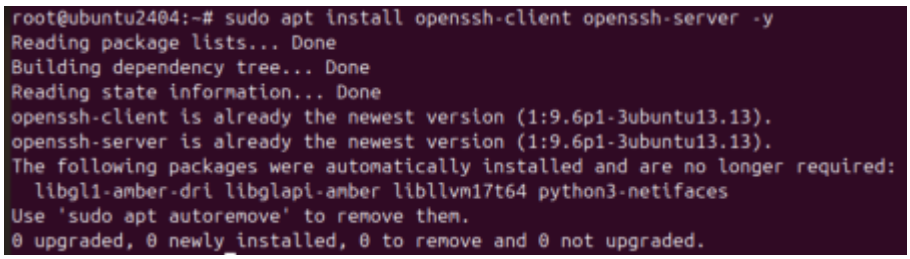
Para poder mantener las conexiones desde el exterior con el servidor de UBUNTU-ELK STACK procedemos a instalar el servicio de SSH, conexión segura.

Ejecutamos el comando :

```
sudo apt install openssh-client openssh-server -y
```

Figura 29

Instalación del cliente OpenSSH



Iniciamos el servicio SSH

Con este comando iniciamos el servicio SSH en tu Ubuntu 24.04:

```
sudo systemctl start ssh
```

Figura 30

Inicio de servicio SSH

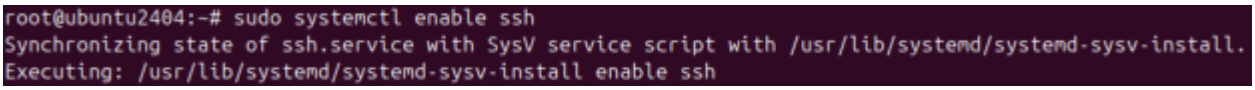
Habilitar el servicio SSH

Habilitamos el servicio SSH en el arranque mediante el comando:

```
sudo systemctl enable ssh
```

Figura 31

Habilitación de SSH



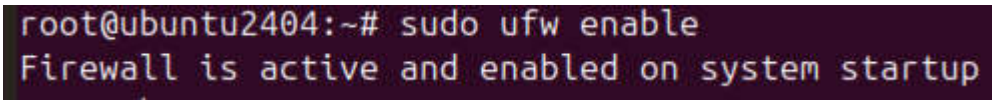
Habilitamos el firewall para SSH.

El siguiente comando es esencial para configurar un firewall para controlar el tráfico de red en el server

```
sudo ufw enable
```

Figura 32

Habilitación de UFW



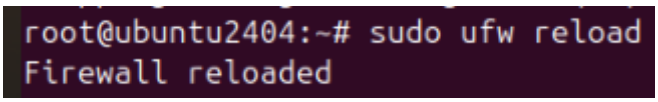
Recargar el firewall.

Para aplicar los cambios actualizados, recargamos el Firewall usando este comando:

```
sudo ufw reload
```

Figura 33

Recarga del firewall



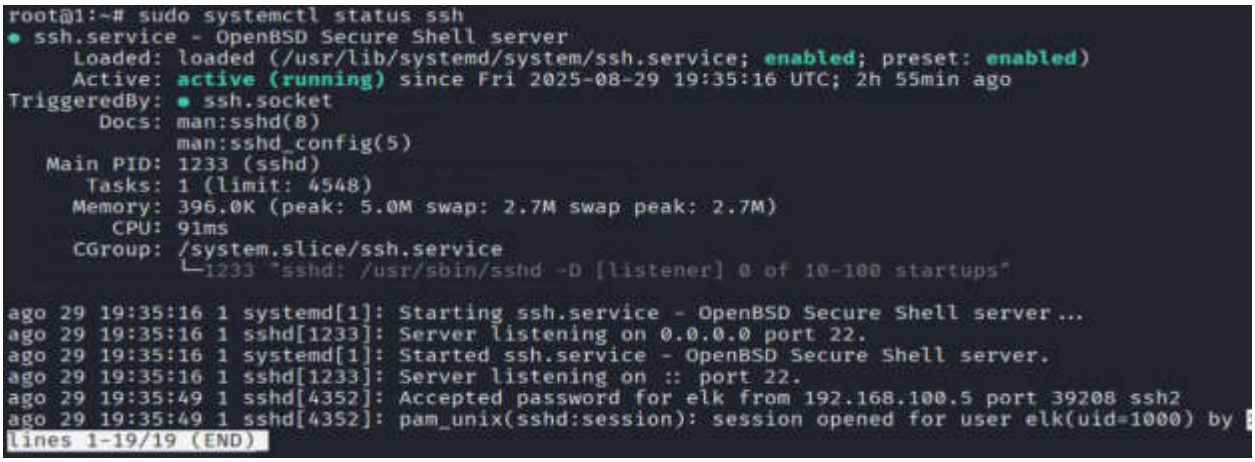
Verificar el estado de SSH

Ahora, podemos verificar el estado SSH existente en nuestro sistema Ubuntu:

```
sudo systemctl status ssh
```

Figura 34

Verificación de estatus del SSH

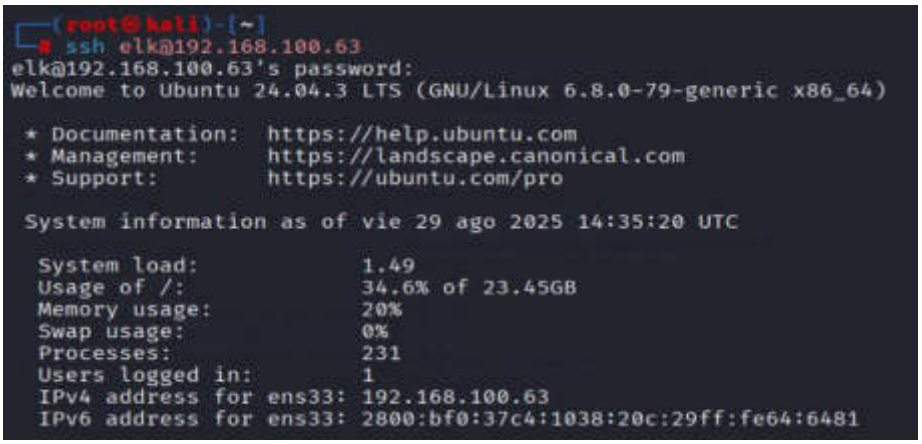


Ahora ya podemos conectarnos remotamente al servidor con la clave (elk) para poder extraer la RAM.

Nos conectamos desde KALI mediante SSH para poder realizar las instalaciones y configuraciones necesarias para levantar nuestro SIEM

Figura 35

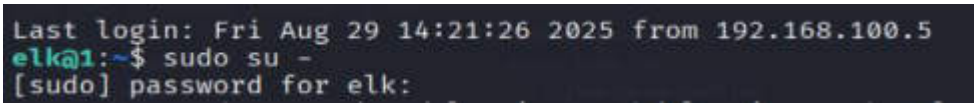
Conexión desde Kali para habilitar SIEM



Nos logeamos y colocamos la clave para root con ello no tendremos inconvenientes al utilizar el sudo

Figura 36

Logeo en Kali.



Instalación y Configuración Del SIEM

Instalamos el Elastic Stack

De forma predeterminada, los tres componentes de la pila ELK no están disponibles en los repositorios aptos de Ubuntu. Por lo tanto, se debe agregar manualmente la lista de fuentes de paquetes de Elastic. Para ello, primero agregamos la clave GPG mediante el comando.

```
wget https://artifacts.elastic.co/GPG-KEY-elasticsearch -O
/etc/apt/keyrings/GPG-KEY-elasticsearch.key
```

Figura 37

Importación manual de paquetes desde Elastic

Agregamos el repositorio Elasticsearch:

`echo "deb [signed-by=/etc/apt/keyrings/elasticsearch.gpg] https://artifacts.elastic.co/packages/8.x/apt stable main" | sudo tee /etc/apt/sources.list.d/elastic-8.x.list`

Figura 38

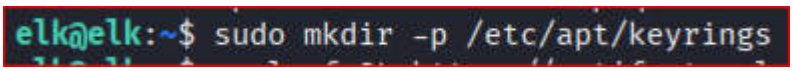
Inclusión del repositorio Elasticsearch

Nos da un error al momento de cargar las llaves para ello realizamos los siguientes pasos:

Creamos un nuevo directorio: `mkdir -p /etc/apt/keyrings`, y extraemos las claves al nuevo directorio.

Figura 39

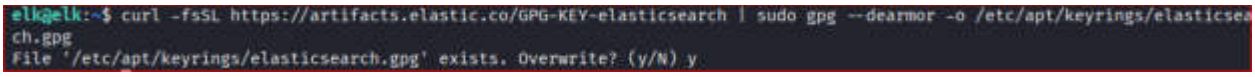
Creación de directorio /etc/apt/keyrings



`curl -fsSL https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo gpg --dearmor -o /etc/apt/keyrings/elasticsearch.gpg`

Figura 40

Ejecución de comando curl -fsSL

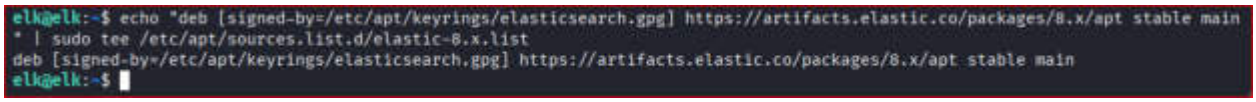


Agregamos el repositorio APT de elastick 8

```
echo "deb [signed-by=/etc/apt/keyrings/elasticsearch.gpg]
https://artifacts.elastic.co/packages/8.x/apt stable main" | sudo tee
/etc/apt/sources.list.d/elastic-8.x.list
```

Figura 41

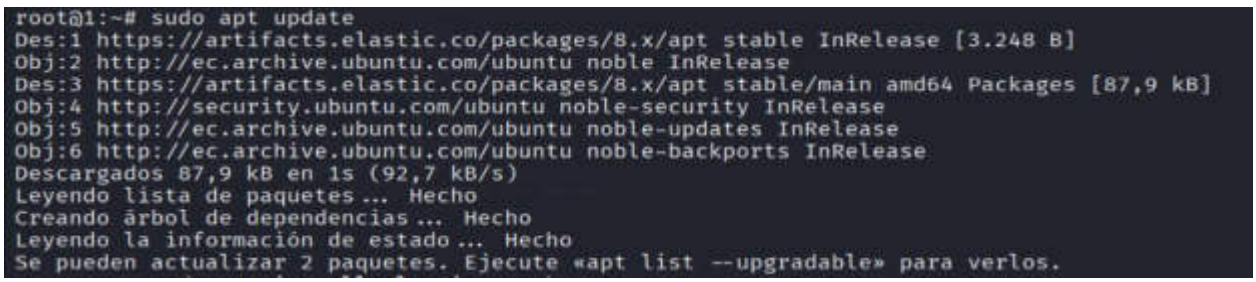
Inclusión del repositorio APT de elastick 8



Realizamos la actualización respectiva

Figura 42

Actualización para inclusión de elastick 8



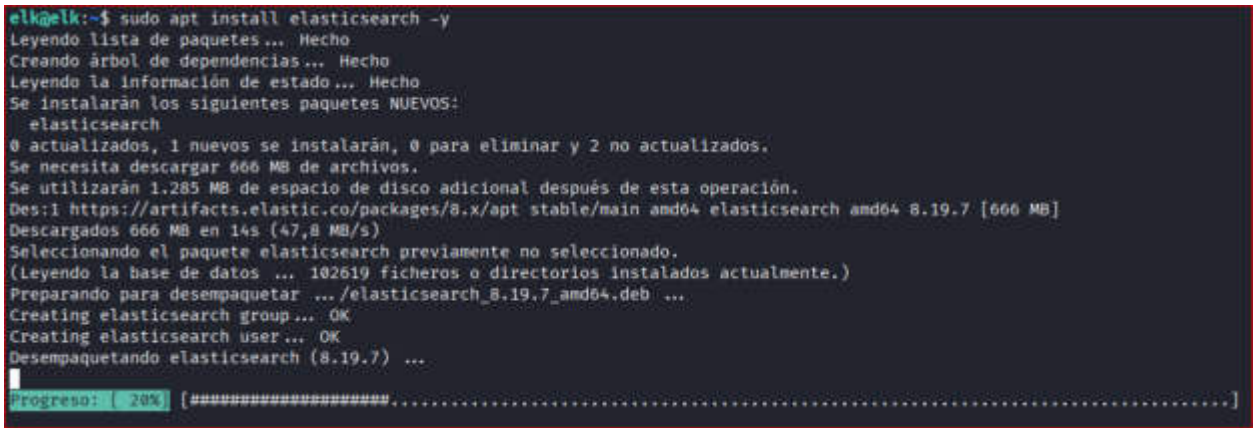
Instalamos Elasticsearch

Para instalar Elasticsearch, ejecutamos el comando:

```
sudo apt install elasticsearch -y
```

Figura 43

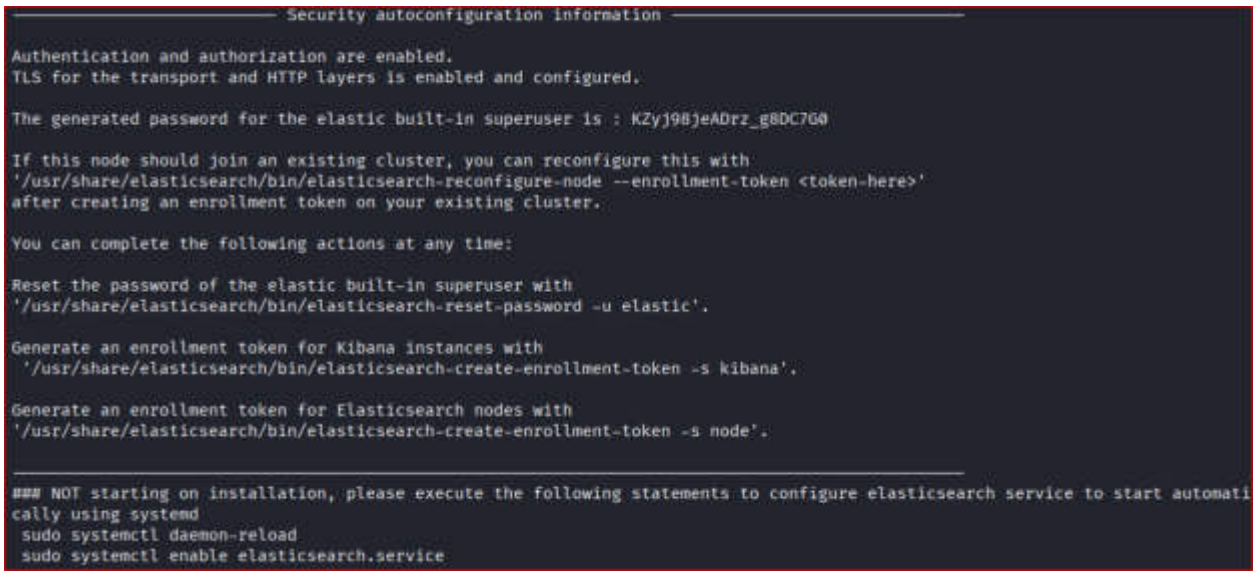
Instalación de Elasticsearch



Una vez completada la instalación, la información de autoconfiguración de seguridad se imprimirá en la pantalla.

Figura 44

Autoconfiguración de seguridad



Una vez completada la instalación, iniciamos el servicio Elasticsearch ejecutando los siguientes comandos.

```
sudo systemctl daemon-reload

sudo systemctl start elasticsearch
```

Figura 45

Inicio del servicio Elasticsearch

Habilitamos el servicio Elasticsearch para que se inicie automáticamente después de la instalación.

```
sudo systemctl enable elasticsearch
```

Figura 46

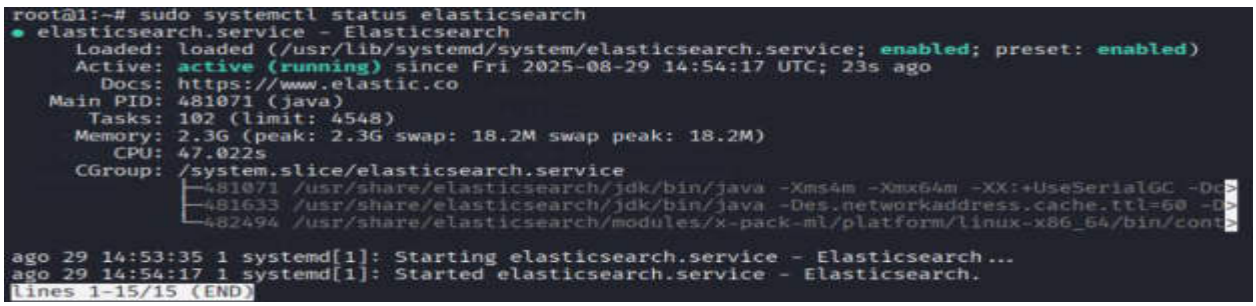
Habilitación del servicio Elasticsearch

Para confirmar que Elasticsearch se está ejecutando, ejecutamos el comando:

```
sudo systemctl status elasticsearch
```

Figura 47

Verificación de estatus de Elasticsearch



Configuramos Elasticsearch

Antes de continuar, se debe de realizar algunas modificaciones en Elasticsearch, accedemos al archivo de configuración de Elasticsearch.

```
sudo nano /etc/elasticsearch/elasticsearch.yml
```

Para comenzar, debemos darle un nombre descriptivo a nuestro clúster descomentando la directiva cluster.name y proporcionando su nombre preferido.

Aquí llamamos a nuestro grupo sample-cluster.

```
cluster.name: proyectoPBL-G7
```

Figura 48

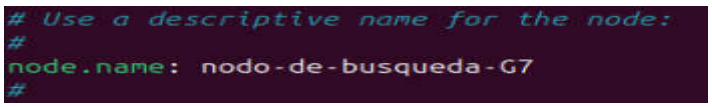
Creación de grupo sample-cluster

A continuación, le damos un nombre descriptivo para el nodo descomentando la directiva node.name.

```
node.name: nodo-de-busqueda-G7
```

Figura 49

Asignación del nombre descriptivo para el nodo



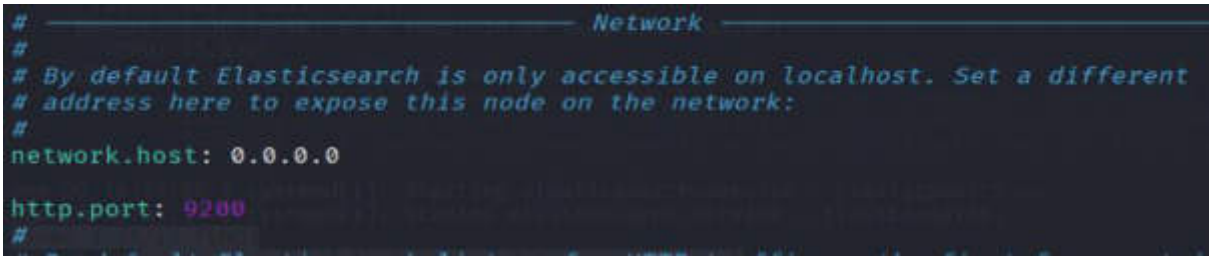
De forma predeterminada, Elasticsearch solo es accesible en el host local. Para que sea accesible desde el exterior, descomentamos y actualizamos el atributo:

`network.host: 0.0.0.0`

`http.port: 9200`, y configuramos el puerto para la conexión externa.

Figura 50

Configuración del puerto para conexión externa



Este apartado del archivo de configuración de Búsqueda elástica (elasticsearch.yml) que menciona para mejorar el rendimiento y la estabilidad de Elasticsearch, especialmente en entornos de producción.

Vamos a:

`bootstrap.memory_lock: true`

Este parámetro sirve para bloquear la memoria del proceso de Elasticsearch en la RAM (evita que el sistema operativo la intercambie al disco con swap).

Por qué bloquear la memoria?

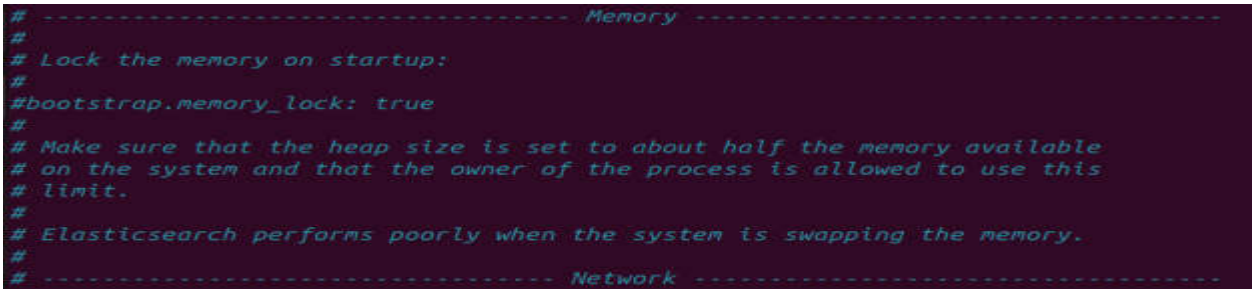
Elasticsearch usa una gran cantidad de memoria para surushap de JVM, si esa memoria es enviada al swap, el rendimiento cae drásticamente.

Bloquearla evita problemas de latencia o caídas cuando el sistema entra en uso intensivo de RAM.

Por ser un ambiente de pruebas la dejamos por defecto.

Figura 51

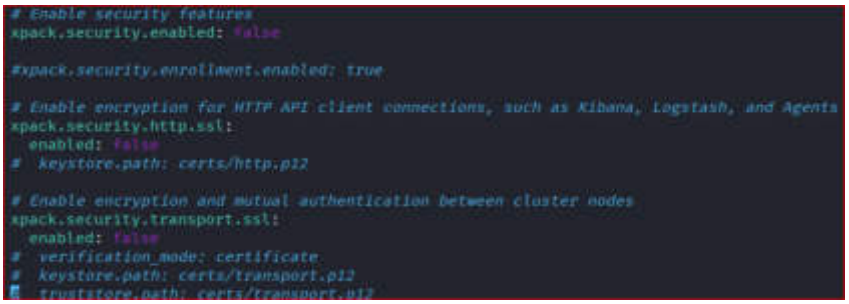
Mensaje de confirmación de ElasticSearch



A continuación, ubicamos la directiva `xpack.security.enabled`: y la configuramos como `false, http ssl y transport` en false, esto solo para pruebas de produccion.

Figura 52

Configuración de flase, hhtp ssl y transport



Guardamos los cambios y salimos del archivo de configuración. Para aplicar los cambios, reiniciamos Elasticsearch:

```
sudo systemctl restart elasticsearch
```

Figura 53

Reiniciado de Estasticsearch

En este apartado nos da error de memoria ya que anteriormente configuramos para que bloquee la memoria conforme se necesite

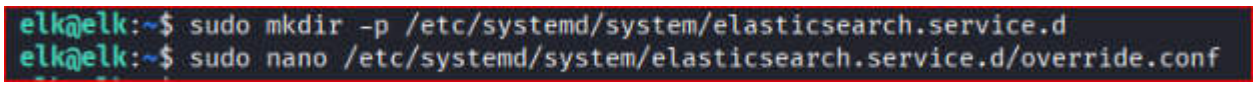
Figura 54

Colocación de contraseña de ELK

Para resolver el inconveniente creamos la ruta y el archivo necesario con los siguientes comandos y editamos el archivo

Figura 55

Creación de ruta y configuración para Elasticsearch



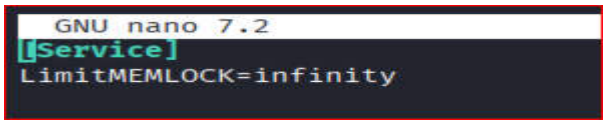
Agregamos las siguientes líneas

```
[Service]

LimitMEMLOCK=infinity
```

Guardamos y cerramos el archivo

Figura 56

Guardado de archivo override luego de su configuración



Reiniciamos los servicios con los siguientes comandos

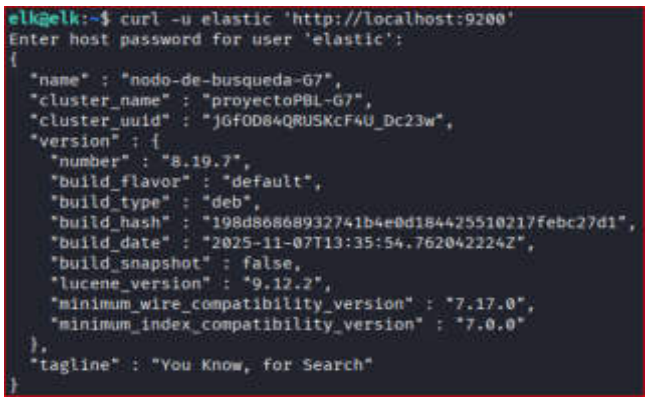
```
sudo systemctl daemon-reexec
sudo systemctl restart elasticsearch
sudo systemctl status elasticsearch
```

Para comprobar si el servicio Elasticsearch se está ejecutando, enviamos una solicitud HTTP utilizando la utilidad Curl como se muestra:

```
curl -u elastic 'http://localhost:9200'
```

Figura 57

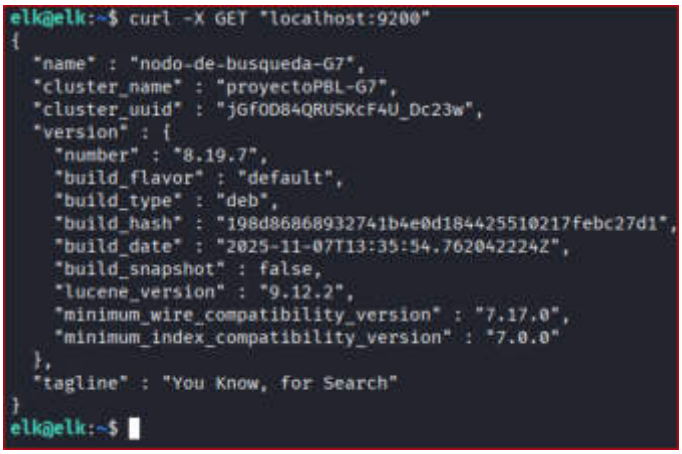
Comprobación de servicio Elasticsearch mediante curl




```
curl -X GET "localhost:9200"
```

Figura 58

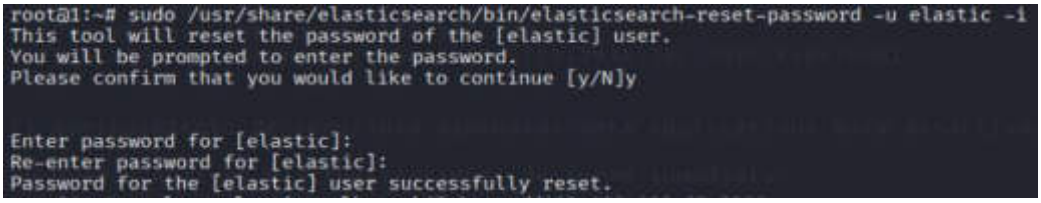
Colocación de parámetros -X GET "localhost:9200"



En esta parte es importante revisar las configuraciones ya que nos da error al correr la aplicación desde el navegador, para ello lo primero que hacemos es resetear la clave

Figura 59

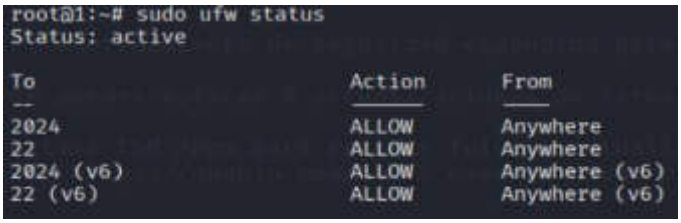
Revisión de configuraciones para Elasticsearch



Luego debemos dar permisos en el firewall al puerto con el que estamos trabajando, primero verificamos el estado

Figura 60

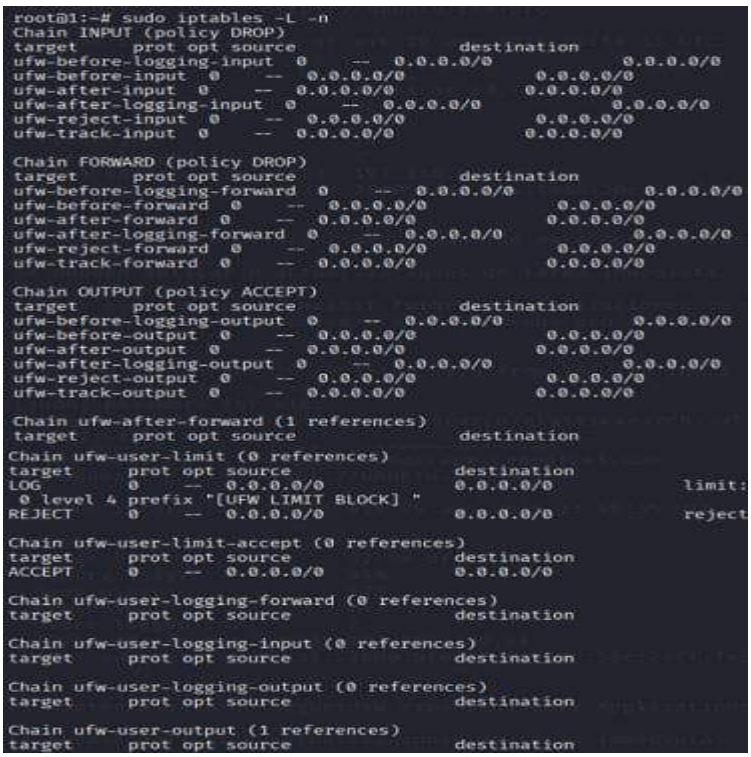
Otorgación de permisos al firewall



Revisamos las IP-TABLES para verificar si en efecto tenemos permiso del firewall

Figura 61

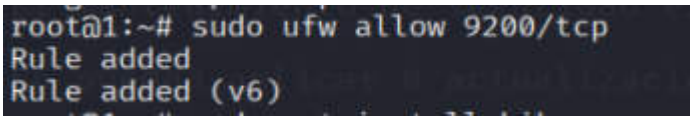
Revisión de IP-TABLES



Agregamos el puerto 9200 al firewall que es el puerto que utiliza ELASTIC

Figura 62

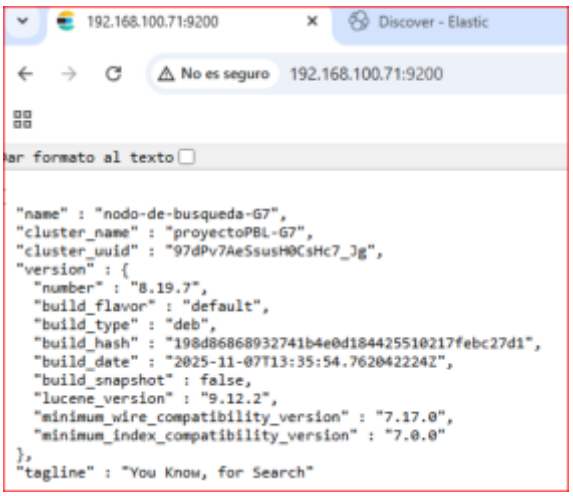
Inclusión de puerto 9200 en el firewall



Verificamos desde el navegador; Usuario y clave `elastic / elkstackG7`

Figura 63

Verificación desde el navegador

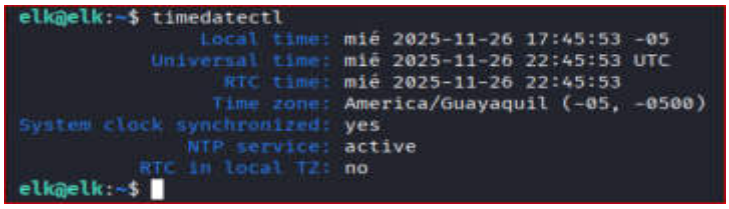


Antes de seguir con las instalaciones y configuraciones es necesario cambiar la zona horaria para que no exista inconvenientes al momento de sincronizar los equipos dentro de la red.

Ingresamos: `sudo timedatectl set-timezone America/Guayaquil`

Figura 64

Cambio de zona horaria en ELK



Instalar Kibana

El siguiente componente para instalar es Kibana. Se trata de una potente herramienta web de visualización y exploración de datos para visualizar registros y series temporales.

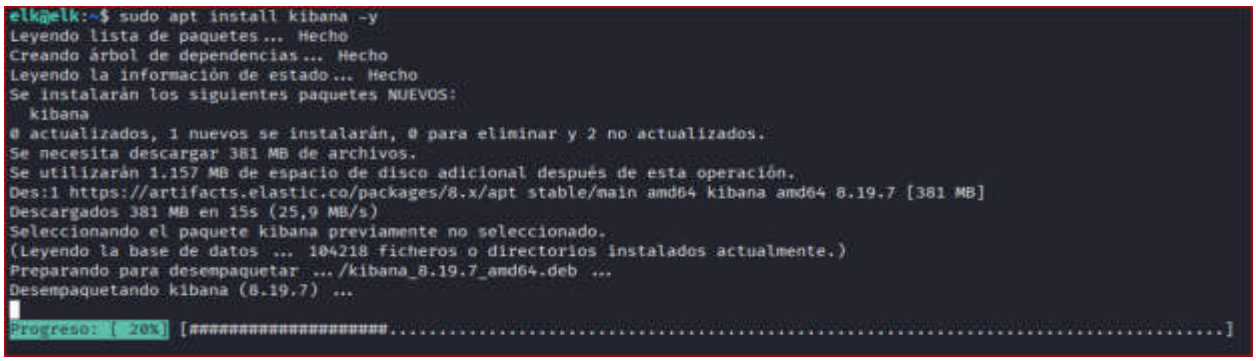
Ofrece paneles potentes y con numerosas funciones, así como una variedad de gráficos como gráficos de barras, gráficos circulares, histogramas, mapas de calor, etc.

Para instalar Kibana, ejecutamos el siguiente comando:

```
sudo apt install kibana -y
```

Figura 65

Instalación de Kibana

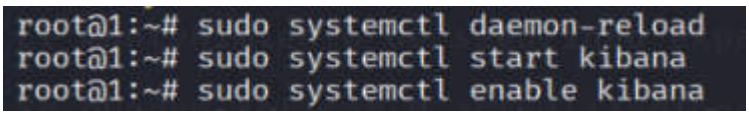


Una vez instalado, iniciamos y habilitamos Kibana para que se inicie automáticamente.

```
sudo systemctl start kibana

sudo systemctl enable kibana
```

Figura 66

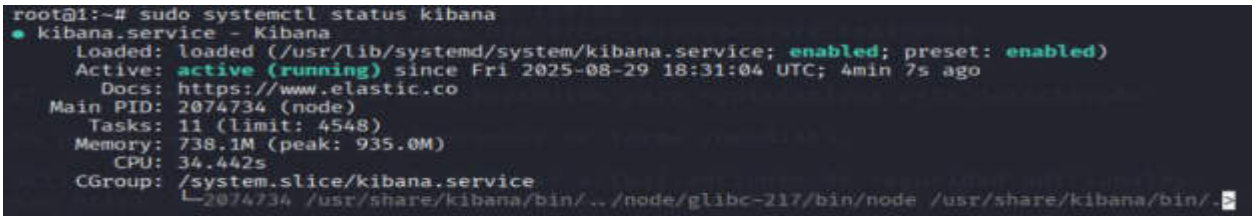
Habilitación de Kibana



Comprobamos el estado de Kibana ejecutando: `sudo systemctl status kibana`

Figura 67

Verificación de estatus de Kibana



Kibana escucha en el puerto TCP 5601 por defecto, configuramos el puerto ya que necesita de la clave de Elastic para su conexión: verificamos `sudo ss -pnltu | grep 5601`

Figura 68

Configuración de Kibana

Configuramos Kibana

Se necesitan algunos ajustes adicionales en Kibana para lograr una experiencia fluida y permitir que funcione con otros componentes u conexiones externas.

Entonces, accedemos al archivo de configuración de Kibana con el comando

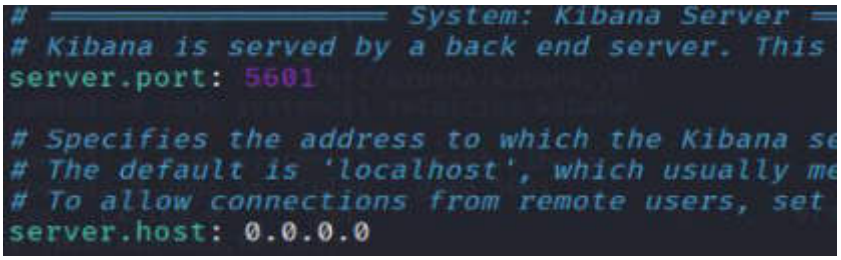
```
sudo nano /etc/kibana/kibana.yml
```

Desmarcamos el comentario de la línea a continuación.

```
Server.port: 5601 y server host 0.0.0.0
```

Figura 69

Configuración de puertos de Kibana

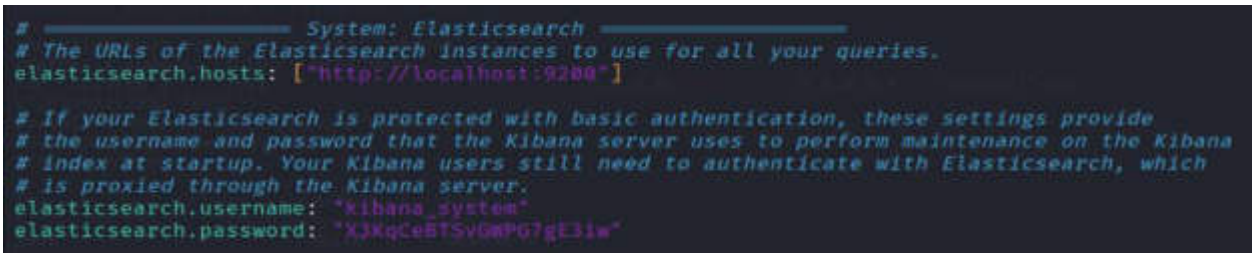


Para permitir conexiones de usuarios remotos, configuramos las siguientes directivas, adicional a esto hay que tener en cuenta que se volvió a generar la clave de Elastic y es la que colocamos en el archivo de configuración para exista comunicación entre las dos aplicaciones.

```
kivana: X3KqCeBTSvGWPG7gE3iw
```

Figura 70

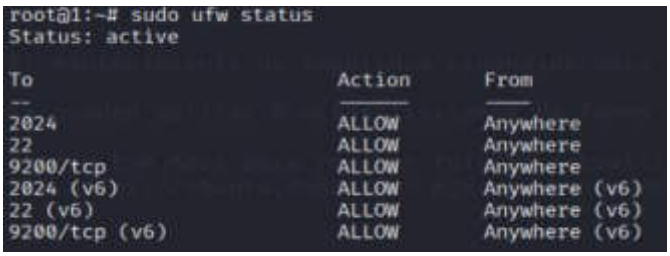
Configuración de directivas para usuarios remotos



Guardamos los cambios y salimos del entorno para aplicar los cambios reiniciamos Kibana, verificamos el estado del firewall para darle asignarle y darle permisos al puerto 5601.

Figura 71

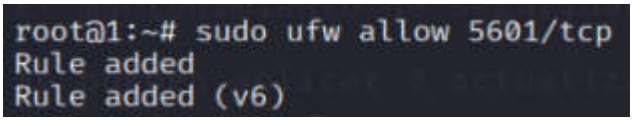
Verificación del estado del firewall para Kibana



Agregamos el puerto 5601 en las IP-TABLES

Figura 72

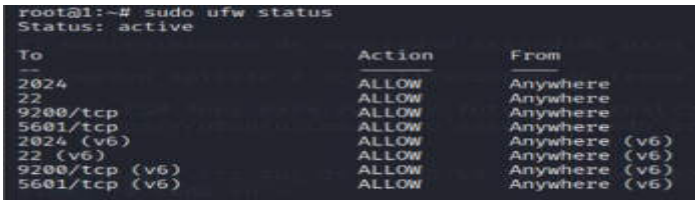
Inclusión del puerto 5601 en IP-TABLES



verificamos nuevamente el estado del firewall

Figura 73

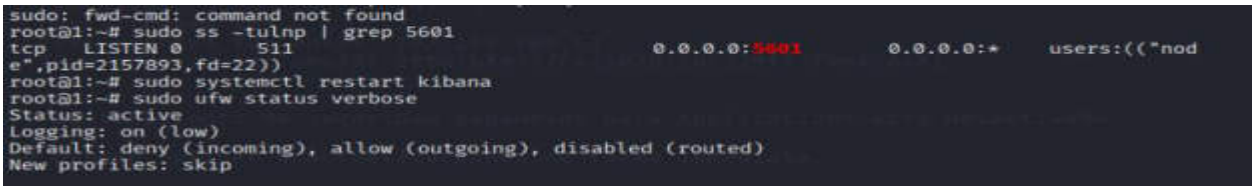
Nueva verificación de firewall para Kibana



Hacemos que el puerto sea permanente y comprobamos que este a la escucha

Figura 74

Configuración del puerto 5601 como permanente

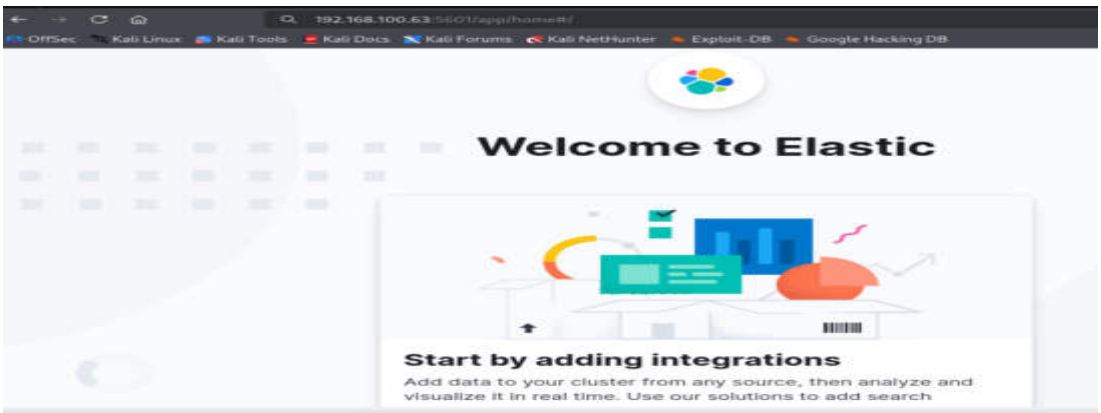


Para acceder a Kibana desde el navegador web, colocamos la siguiente URL.

`https://192.168.100.71:5601`

Figura 75

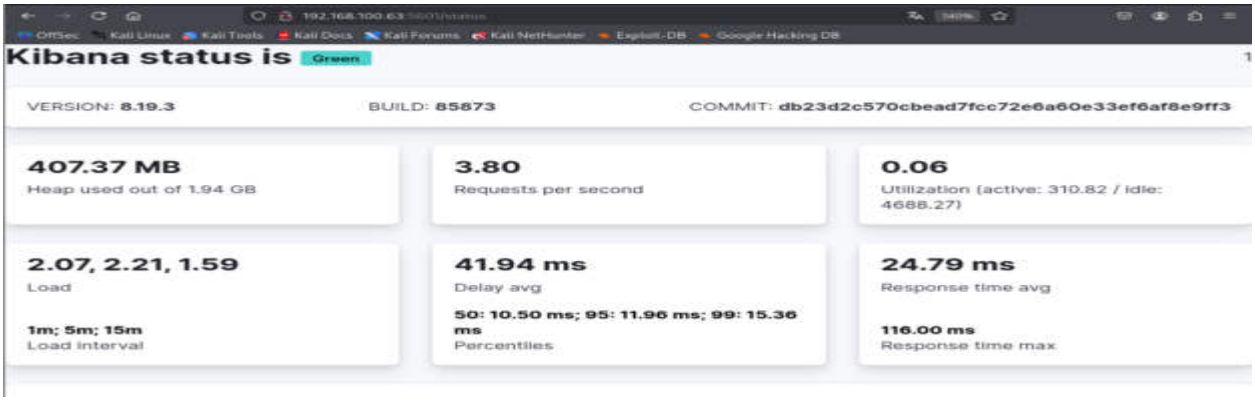
Acceso a Kibana en el navegador web



Además, se puede tener una visión general de su estado y servicios disponibles visitando la siguiente URL. `http://192.168.100.71:5601/status`

Figura 76

Verificación de estado y servicios disponibles de Kibana en el navegador web

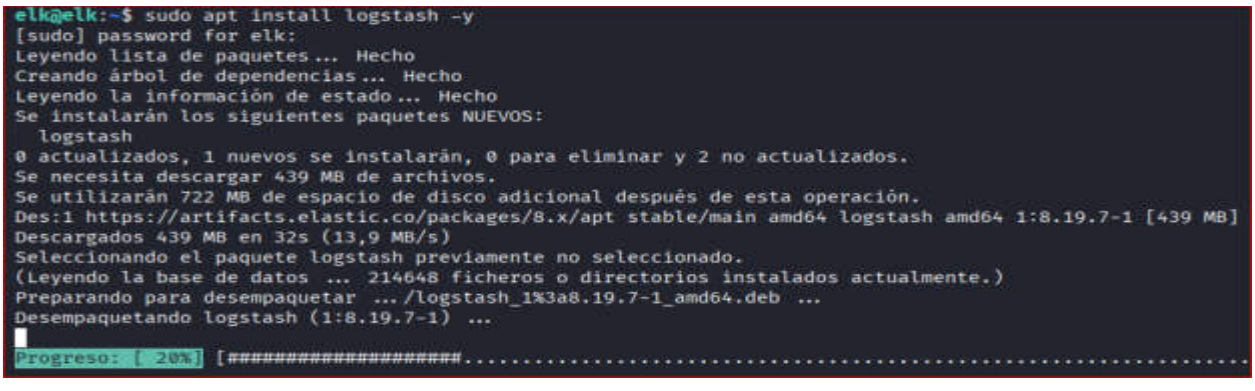


Instalar Logstash

El siguiente componente para instalar será Logstash, esta canalización procesa y envía datos a Elasticsearch para su indexación y almacenamiento, para instalar Logstash, ejecutamos el siguiente comando: `sudo apt install logstash -y`

Figura 77

Instalación de Logstash

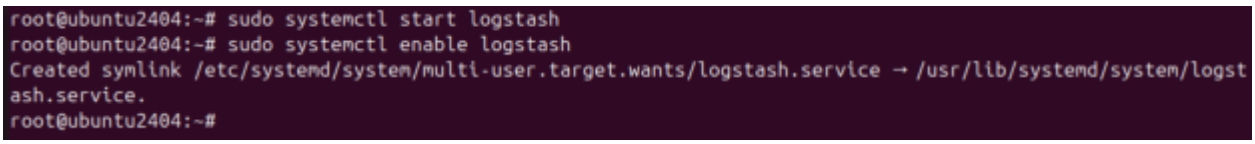


Una vez instalado, inicializamos y habilitamos Logstash ejecutando los siguientes comandos

```
sudo systemctl start logstash

sudo systemctl enable logstash
```

Figura 78

Habilitación de Logstash

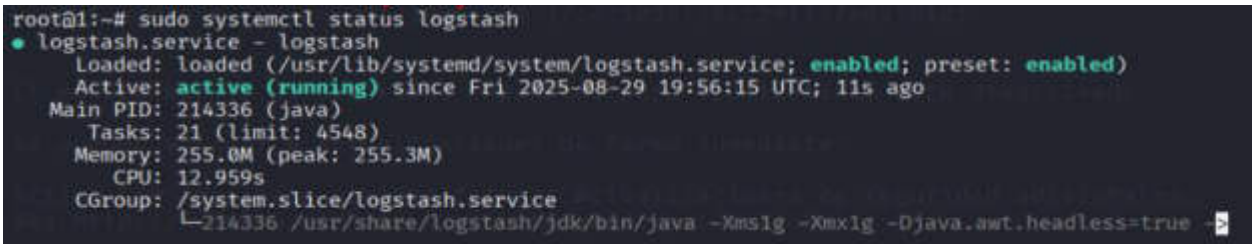


Comprobamos el estado del almacenamiento de registros, ejecutando el comando:

```
sudo systemctl status logstash
```

Figura 79

Verificación de estado de Logstash



Configurar Logstash

Como se mencionó anteriormente, Logstash es una canalización que analiza y procesa datos y los envía al destino deseado, en este caso, Elasticsearch.

Una canalización consta de dos elementos obligatorios: entrada y salida. El elemento de filtro es opcional. El complemento de entrada toma o ingiere datos del origen. Los filtros son complementos intermediarios que procesan los datos mientras el complemento de salida los envía y escribe en un destino.

A continuación, creamos un archivo de configuración llamado 02-beats-input.conf.

Aquí configuramos la entrada de Filebeat:

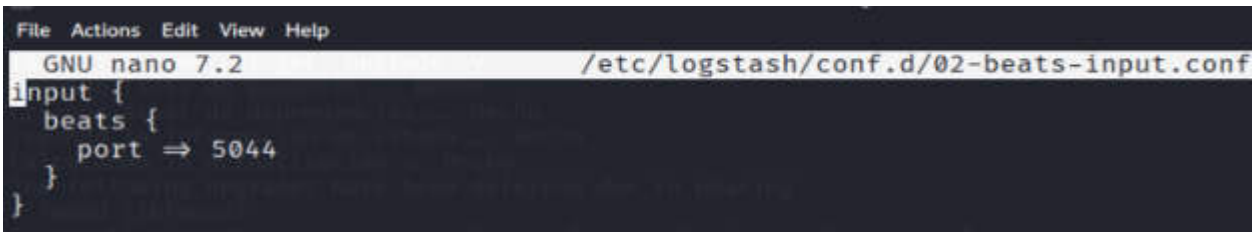
```
sudo nano /etc/logstash/conf.d/02-beats-input.conf
```

Agregamos las siguientes líneas de código:

```
input {
  beats {
    port => 5044
  }
}
```

Figura 80

Configuración del puerto de Logstash



A continuación, creamos un archivo de configuración llamado 30-elasticsearch-output.conf:

```
sudo nano /etc/logstash/conf.d/30-elasticsearch-output.conf
```

La salida configura Logstash para almacenar los datos de Beats en Elasticsearch, ejecutándose en localhost:9200.

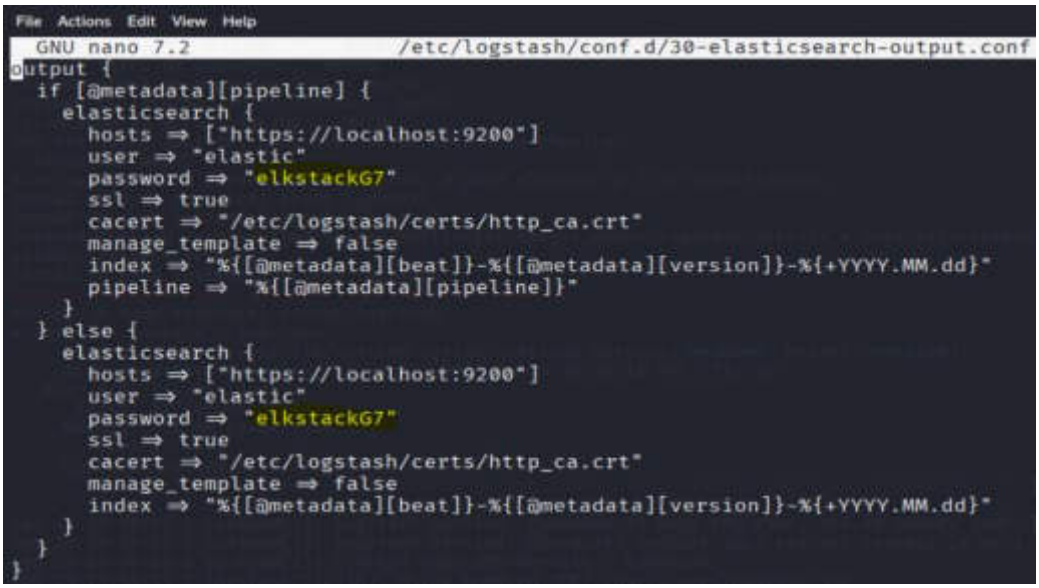
Usamos Beats como índice, filebeat será compatible con Beats.

```
output {
  if [@metadata][pipeline] {
    elasticsearch {
      hosts => ["https://localhost:9200"]
      user => "elastic"
      password => "elkstackG7"
      ssl_enabled => true
      ssl_certificate_authorities =>
["/etc/logstash/certs/http_ca.crt"]
      manage_template => false
      index => "%{[@metadata][beat]}-%{[@metadata][version]}-
%{+YYYY.MM.dd}"
      pipeline => "%{[@metadata][pipeline]}"
    }
  } else {
    elasticsearch {
      hosts => ["https://localhost:9200"]
      user => "elastic"
      password => "elkstackG7"
      ssl_enabled => true
      ssl_certificate_authorities =>
["/etc/logstash/certs/http_ca.crt"]
      manage_template => false
```

```
            index => "%{[@metadata][beat]}-%{[@metadata][version]}-
%{+YYYY.MM.dd}"
          }
        }
      }
```

Figura 81

Configuraciones adicionales de Logstash



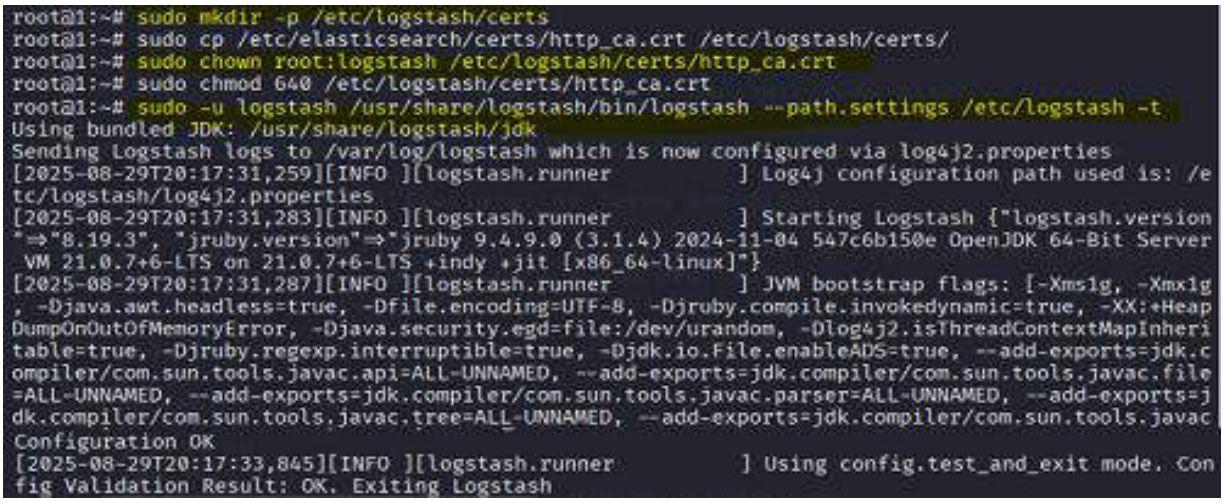
Guardamos y salimos, para probar la configuración de Logstash, ejecutamos el comando:

```
sudo -u logstash /usr/share/logstash/bin/logstash --path.settings
/etc/logstash -t
```

En primera instancia nos da error en la configuración ya que no encuentra la llave, para ello creamos un nuevo repositorio, copiamos las claves a este repositorio, eliminamos las claves anteriores, damos permisos de lectura y lo volvemos a ejecutar.

Figura 82

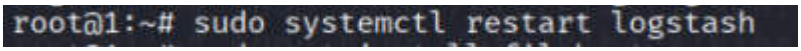
Verificación inicial de error en Logstash



La salida consta de un bloque de líneas de código con la cadena Resultado de validación de configuración: OK, Para que los cambios surtan efecto, debemos reiniciar logstash con el comando: `sudo systemctl restart logstash`

Figura 83

Reiniciado del servicio de Logstash



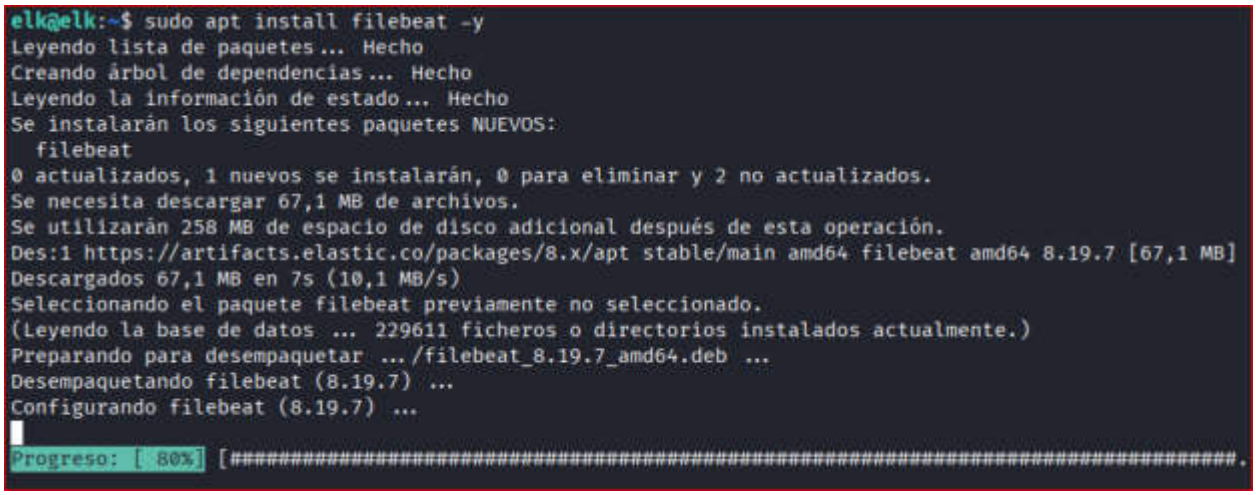
Instalar y configurar Filebeat

El último componente para instalar es Filebeat, este es un gestor de envíos ligero que reenvía datos de registro centralizados a Elasticsearch o Logstash, según su configuración.

Para instalar Filebeat, ejecutamos el comando: `sudo apt install filebeat -y`

Figura 84

Instalación de Filebeat



A continuación, configuraremos Filebeat para enviar datos a Logstash, accedemos al archivo de configuración de Filebeat: `sudo nano /etc/filebeat/filebeat.yml`

Configuraremos Filebeat para que envíe los datos directamente a Logstash para su procesamiento en lugar de Elasticsearch. Por lo tanto, deshabilitaremos la salida de Elasticsearch, para ello, localizamos la sección `output.elasticsearch` y comentamos las siguientes líneas:

Figura 85

Configuración de Filebeat para envío de datos a Logstash

A continuación, configuraremos Filebeat para conectarse a Logstash en el servidor Elastic Stack en el puerto 5044, descomentamos las líneas `output.logstash:` y `hosts: ["localhost:5044"]`.

Figura 86

Configuración de Filebeat para conexión a Logstash

Una vez hecho esto, guardamos los cambios y salimos.

Los módulos de Filebeat amplían la funcionalidad de Filbeat. Aquí, habilitaremos el módulo del sistema que recopila y analiza los registros generados por el servicio de registro del sistema, para obtener una lista de todos los módulos disponibles, ejecute el comando:

```
sudo filebeat modules list
```

Figura 87

Verificación de módulos activos e inactivos en Filebeat

Para habilitar el módulo del sistema, ejecutamos el comando:

```
sudo filebeat modules enable system
```

Figura 88

Habilitación de módulo de sistema en Filebeat

A continuación, se debe de configurar las canalizaciones de ingesta de Filebeat.

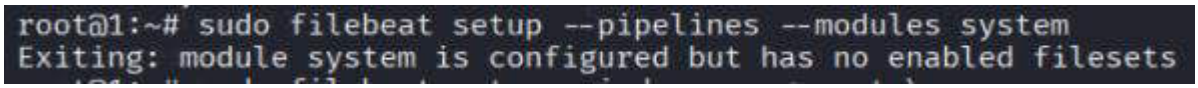
Estas canalizaciones analizan los datos de registro antes de pasarlos a través de Logstash a Elasticsearch.

Ejecutamos el siguiente comando para cargar la canalización de ingesta del módulo del sistema:

```
sudo filebeat setup --pipelines --modules system
```

Figura 89

Configuración de canalización de ingesta en Filebeat



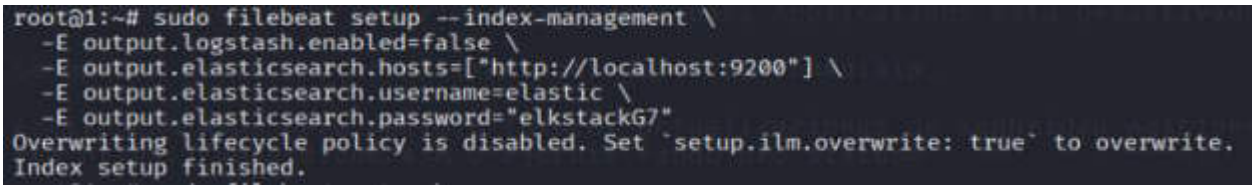
A continuación, se debe de cargar la plantilla de índice en Elasticsearch. Un índice es simplemente un conjunto de documentos con características similares. Para cargar la plantilla, ejecutamos el siguiente comando:

```
sudo filebeat setup --index-management \
```

```
  -E output.logstash.enabled=false \
  -E output.elasticsearch.hosts=["http://localhost:9200"] \
  -E output.elasticsearch.username=elastic \
  -E output.elasticsearch.password="elkstackG7"
```

Figura 90

Configuración de plantilla de índice en Elasticsearch



Filebeat proporciona paneles de Kibana de ejemplo de forma predeterminada para visualizar datos de Filebeat en Kibana.

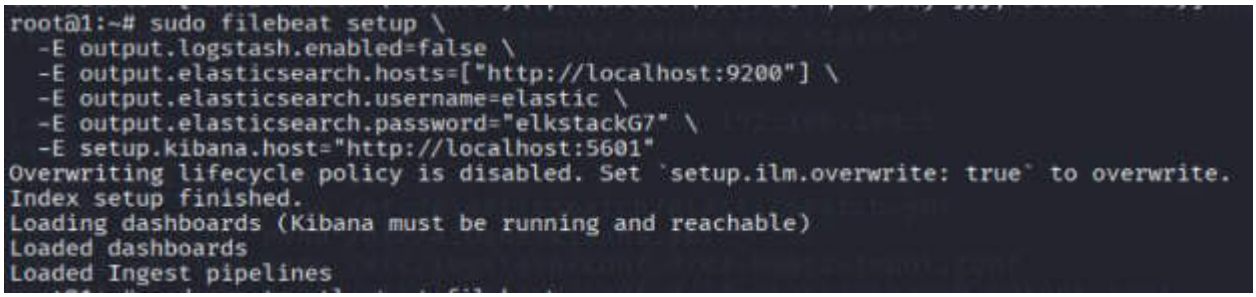
Por lo tanto, antes de usar los paneles, es fundamental crear el patrón de índice y cargarlos en Kibana.

Para ello, ejecutamos el comando:

```
sudo filebeat setup \
  -E output.logstash.enabled=false \
  -E output.elasticsearch.hosts=["http://localhost:9200"] \
  -E output.elasticsearch.username=elastic\
  -E output.elasticsearch.password="elkstackG7" \
  -E setup.kibana.host="http://localhost:5601"
```

Figura 91

Creación del patrón de índice y carga en Kibana



Inicializamos y habilitamos Filebeat.

```
sudo systemctl start filebeat

sudo systemctl enable filebeat
```

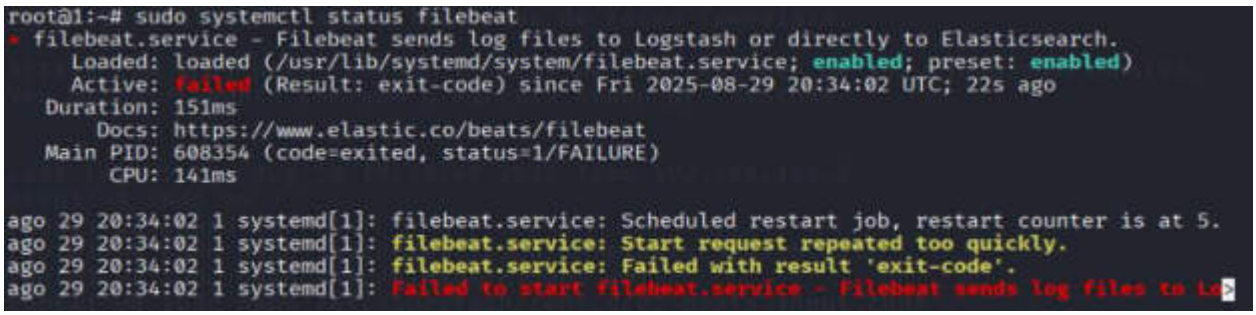
Figura 92

Habilitación del servicio Filebeat

Comparamos su estado `sudo systemctl status filebeat`

Figura 93

Verificación de estado del servicio Filebeat



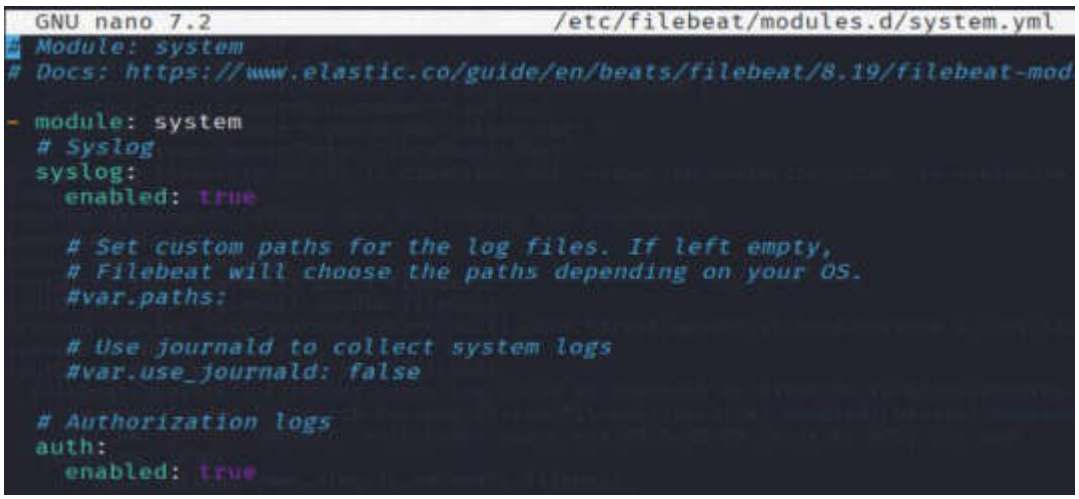
Nos da error ya que no tenemos habilitado el módulo de system

Editamos el módulo `/etc/filebeat/modules.d/system.yml` y colocamos

`enabled: True`

Figura 94

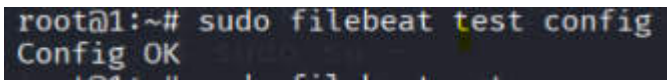
Habilitación de módulo system.yml



Realizamos un test de configuración: `sudo filebeat test config`

Figura 95

Test de configuración de Filebeat



Verificamos la configuración: `sudo filebeat setup`

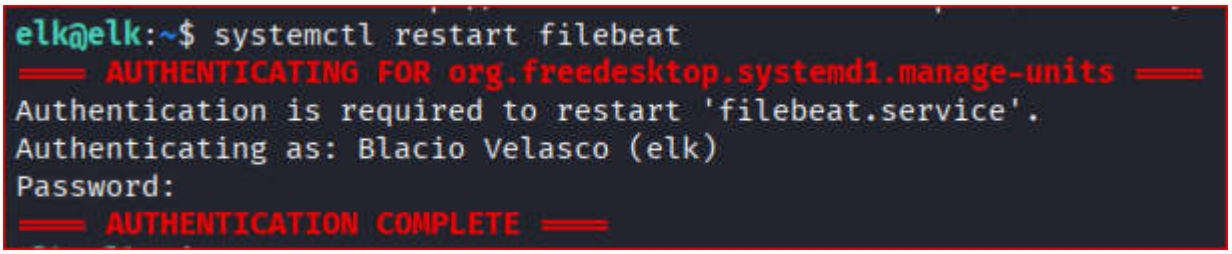
Figura 96

Verificación de configuración en Filebeat Setup

Reiniciamos el servicio: `sudo systemctl restart filebeat`

Figura 97

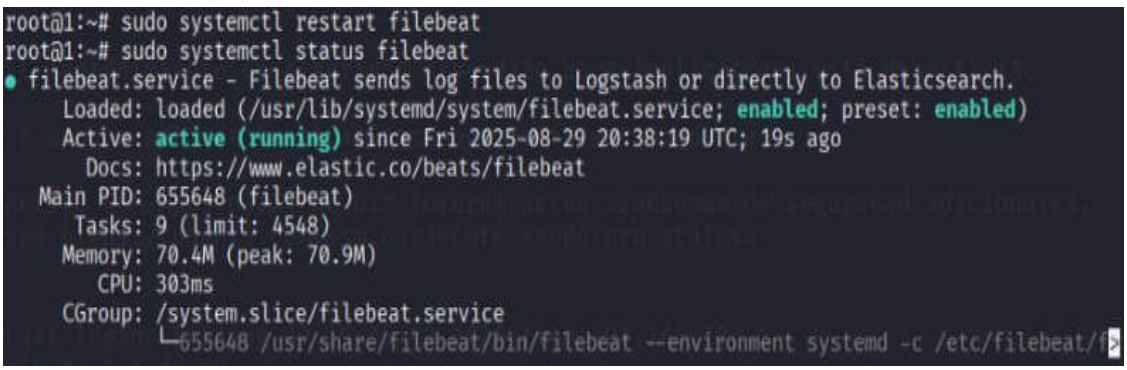
Reinicio del servicio de Filebeat



Comparamos nuevamente su estado: `sudo systemctl status filebeat`

Figura 98

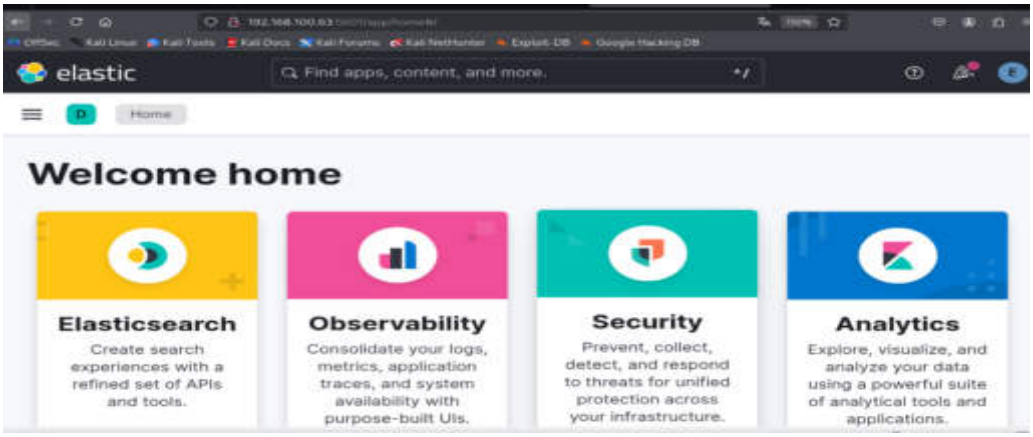
Comprobación nueva del servicio de Filebeat luego de su reinicio



Con esto se comprueba que hemos configurado Filebeat correctamente y debería enviar los datos de registro a Logstash para su procesamiento y, finalmente, a Elasticsearch para su indexación.

Figura 99

Comprobación en navegador web de funcionamiento de Filebeat

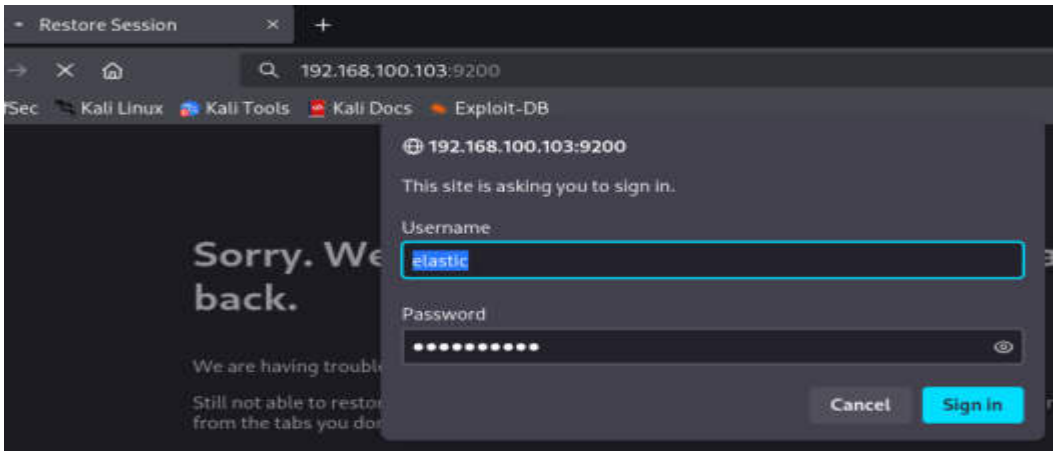


Configuración de KIBANA

Para configurar nuestro SIEM y que comience a monitorizar la red y los diferentes eventos es necesario configurar un usuario el cual nos va a permitir realizar las auditorias dentro del SIEM para ello ingresamos al navegador dentro del equipo de KALI y colocamos la dirección IP del servidor de ELKSTACK (192.168.100.103:9200), no va a pedir el usuario y la contraseña que ya fueron configurados con anterioridad.

Figura 100

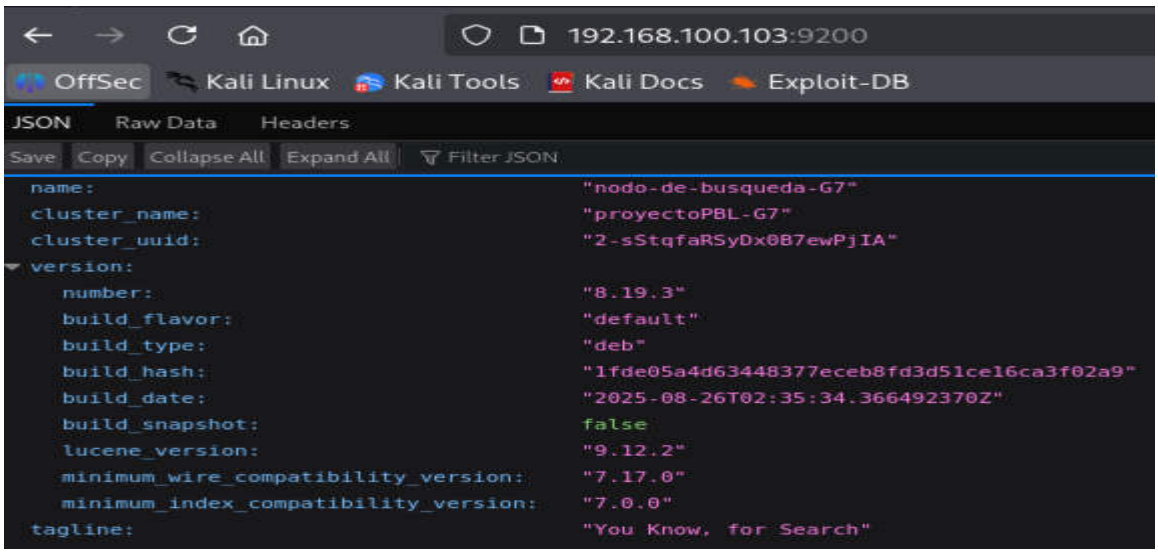
Ventana de ingreso de credenciales en el SIEM



Le damos clic en SIGN IN y nos carga la página principal del SIEM con sus configuraciones iniciales.

Figura 101

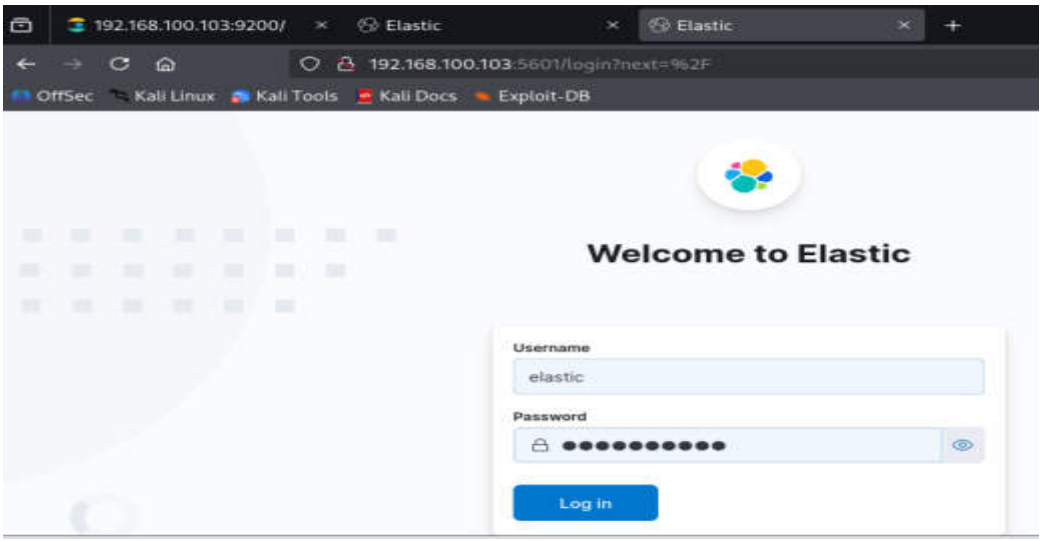
Página principal del SIEM con configuraciones iniciales



De la misma manera ingresamos a KIBANA (192.168.100.103:5601)

Figura 102

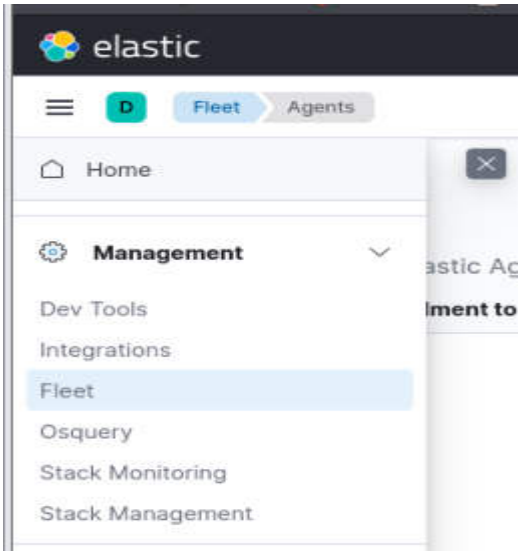
Ingreso desde el navegador web en Kibana



Damos clic en LOG IN y ya podemos ingresar a la página de configuraciones del SIEM, y escogemos la opción Management-Fleet

Figura 103

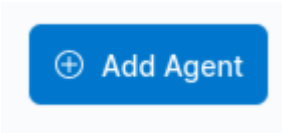
Página de configuraciones del SIEM



Aquí vamos a configurar nuestro usuario (AGENTE) de manera predeterminada, le damos clic en Add Agent

Figura 104

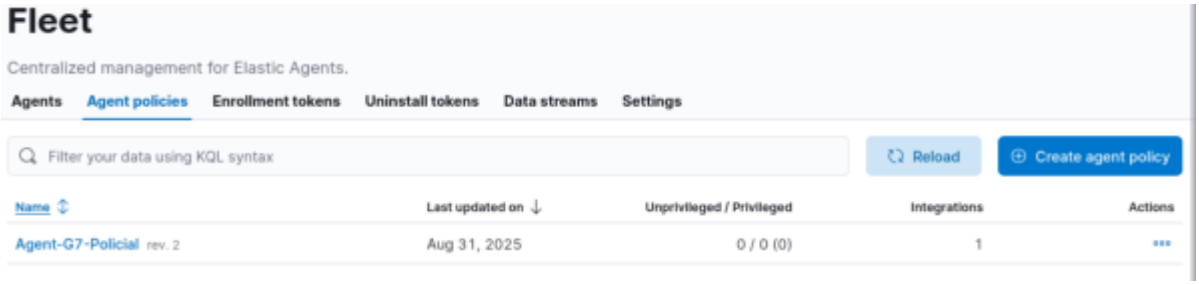
Botón Add Agent en configuración del SIEM



Vamos a crear nuestro Agente llamado Agent-G7-Policial le dejemos por defecto

Figura 105

Creación del agente Agent-G7-Policial



Al momento de crear nuestro agente KIBANA nos da un log con las configuraciones respectivas las mismas que deben de ser cargadas en el archivo de configuración de KIBANA

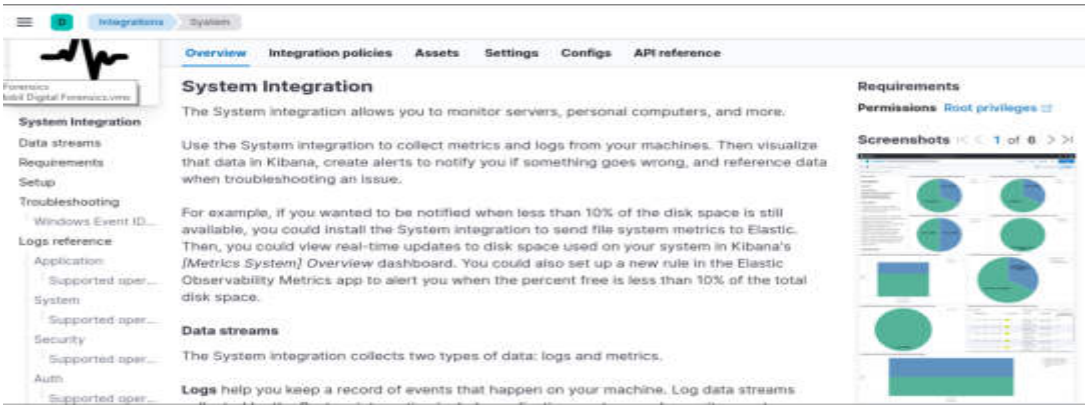
El archivo de configuración lo podemos encontrar en:

`http://192.168.100.71:5601/app/integrations/detail/system-2.5.4/configs`

Una vez creado y configurado nuestro Agente ya comienza a monitorear la RED

Figura 106

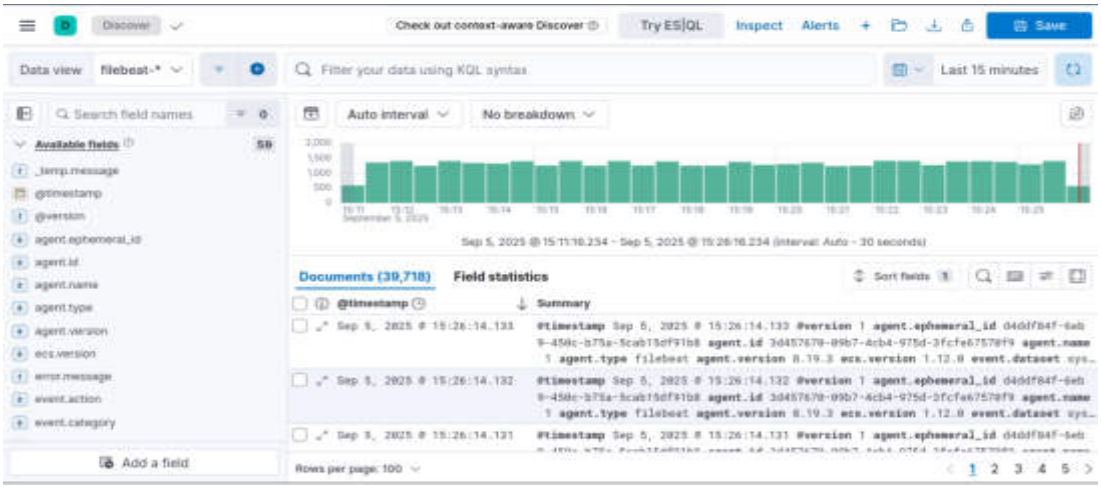
Monitoreo de red en el SIEM



El SIEM comienza a crear los primeros logs, para el Análisis forense procedemos a realizar diversos tipos de ataques al SIEM para poder crear registros de posibles intrusos que intentan acceder a él y también vamos a generar logs con ruido interno.

Figura 107

Primeros logs en el SIEM



En los equipos de Windows10 y de Ubuntu procedemos a instalar las aplicaciones para la recolección de logs, red, etc y posteriormente estos serán enviados al SIEM para su análisis, en el equipo de Ubuntu que es la víctima se instalan:

```
Filebeat → logs del sistema (auth, sudo, syslog)
Auditd (auditbeat) → actividad de seguridad (sudo, comandos, procesos)
Packetbeat → tráfico de red (lo que pasa EN y ENTRE equipos)
Osquery (inventario / cambios en sistema)
```

En la máquina Windows 10 víctima se Instala:

```
Winlogbeat → logs de seguridad (logon, hacking attempts)
Packetbeat → análisis de tráfico
Osquery (inventario / cambios en sistema)
Filebeat → logs del sistema (auth, sudo, syslog)
```

CONFIGURACIÓN PARA UBUNTU VÍCTIMA

Filebeat en Ubuntu

Editamos el archivo de configuración:

```
sudo nano /etc/filebeat/filebeat.yml
```

Activamos los módulos:

```
filebeat modules enable system
filebeat modules enable auditd
filebeat modules enable apache2
```

Salida DIRECTO a Elasticsearch:

```
output.elasticsearch:
  hosts: ["http://192.168.100.71:9200"]
```

Reiniciamos:

```
sudo systemctl enable --now filebeat
```

Packetbeat en Ubuntu

Editamos el archivo de configuración:

```
sudo nano /etc/packetbeat/packetbeat.yml
```

Descomentamos todas las interfaces:

```
packetbeat.interfaces.device: any
```

Salida:

```
output.elasticsearch:
  hosts: ["http://192.168.100.71:9200"]
```

Reiniciamos:

```
sudo systemctl enable --now packetbeat
```

```
Auditbeat en Ubuntu
auditbeat modules enable auditd
```

Salida:

```
output.elasticsearch:
```

```
  hosts: ["http:// 192.168.100.71:9200"]
```

Reiniciamos:

```
sudo systemctl enable --now auditbeat
```

CONFIGURACIÓN PARA WINDOWS 10 VÍCTIMA

Instalamos Winlogbeat

Habilitamos el archivo de configuración en winlogbeat.yml:

output.elasticsearch:

```
  hosts: ["http://192.168.100.71:9200"]
```

Activar logs:

```
winlogbeat.event_logs:
  - name: Security
  - name: System
  - name: Application
```

Instalamos como servicio:

```
winlogbeat.exe install
Start-Service winlogbeat
```

Packetbeat en Windows es opcional pero útil

Realizamos los mismos pasos para elasticsearch:

```
output.elasticsearch:
  hosts: ["http://192.168.100.71:9200"]
```

Para poder receptar los logs en el SIEM se tuvo que redimensionar el disco en especial el **/** ya que nos quedamos sin espacio por las actualizaciones de seguridad que son necesarias instalarlas, hay que tomar en cuenta que el SIEM genera logs e índices innecesarios lo que causa que el disco se llene conforme se va trabajando en él, para ello se ejecutaron los siguientes comandos:

```
sudo lvextend -l +100%FREE /dev/mapper/ubuntu--vg-ubuntu--lv
sudo resize2fs /dev/mapper/ubuntu--vg-ubuntu--lv
df -h / … /dev/mapper/ubuntu--vg-ubuntu--lv   48G   43G   100G  96% /
```

con este nuevo espacio en disco ya podemos instalar y configurar el agente

Ataque a la Infraestructura del SIEM

Preparamos el laboratorio con KALI-LINUX donde vamos a instalar HYDRA, el mismo que nos va a permitir realizar ataques de denegación de servicios, ruidos y vamos a crear un usuario falso dentro del servidor de UBUNTU, para luego a ese mismo usuario suponiendo que sabemos el usuario y la contraseña de KIBANA le vamos a agregar al SIEM para realizar ataques internos.

Para estos ataques y sobre todo para tener evidencias en el SIEM utilizamos los siguientes comandos y payloads. ($TARGET se le configura con las ips de cada equipo a ser atacado 192.168.100.71/73/74/103, etc)

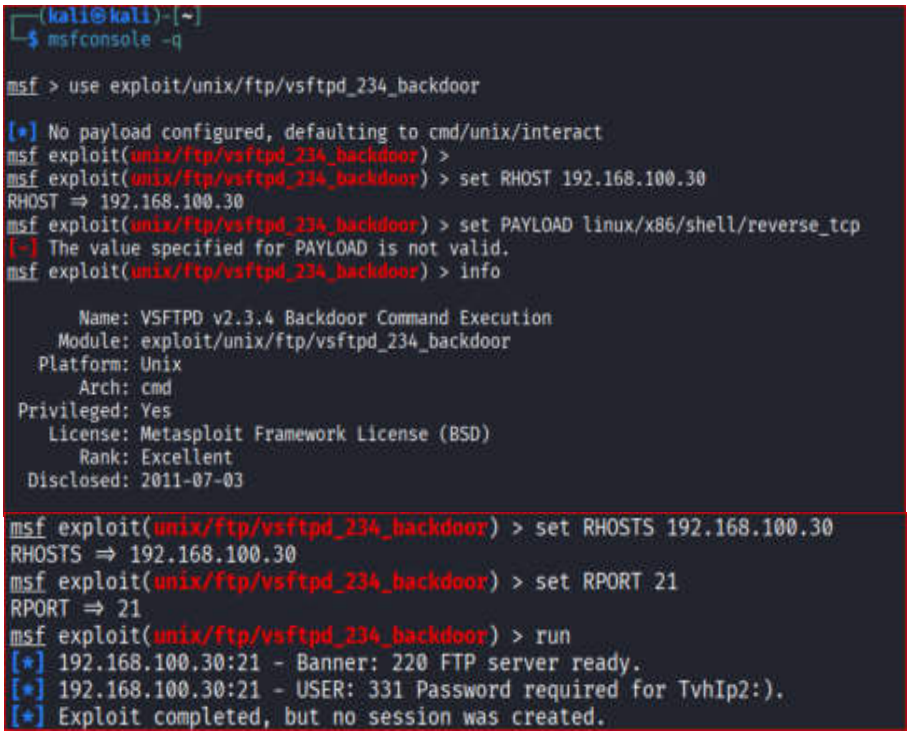
```
sudo nmap -A $TARGET -oN nmap_A_full.txt
```

```
curl -I http:// $TARGET /
nikto -host http:// $TARGET -port 80 -ask no
gobuster dir -u http:// $TARGET -w
/usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -t 40
nmap --script ssl-enum-ciphers -p 443 $TARGET
nmap -sV -sC -Pn $TARGET -oA nmap_scripts
nmap -p 8008,8080,8443 --script vuln $TARGET
hydra -L users.txt -P passwords.txt http-get:// $TARGET:8080/wp-
login.php
hydra -L users.txt -P /usr/share/wordlists/rockyou.txt -t 4 -f
$TARGET mysql -s 3306 -V
hydra -L users.txt -P /usr/share/wordlists/rockyou.txt -t 4 -f
$TARGET mysql -s 3306 -V
nmap -p 22 --script ssh-brute --script-args
userdb=/usr/share/nmap/nselib/data/usernames.lst,passdb=/usr/share/nmap/nse
lib/data/passwords.lst $TARGET
```

Metasploit Framework

Figura 108

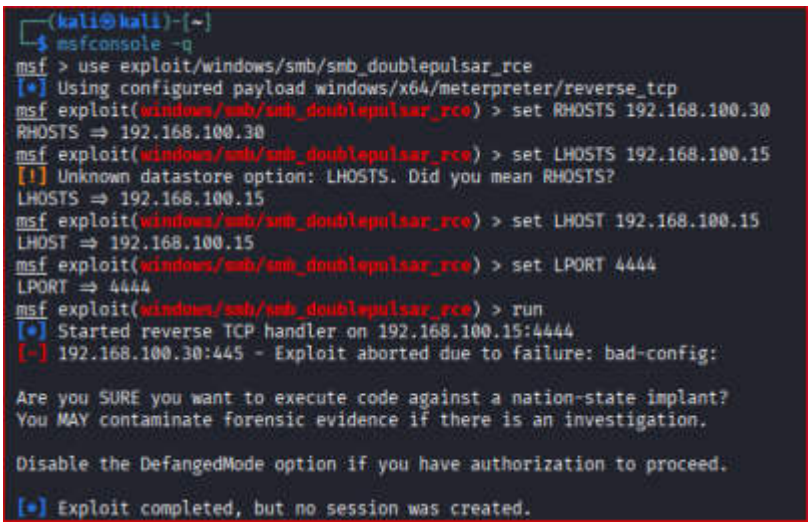
Comandos de Metasploit Framework



Metasploit Framework para smb en Windows

Figura 109

Metasploit Framework para smb en Windows

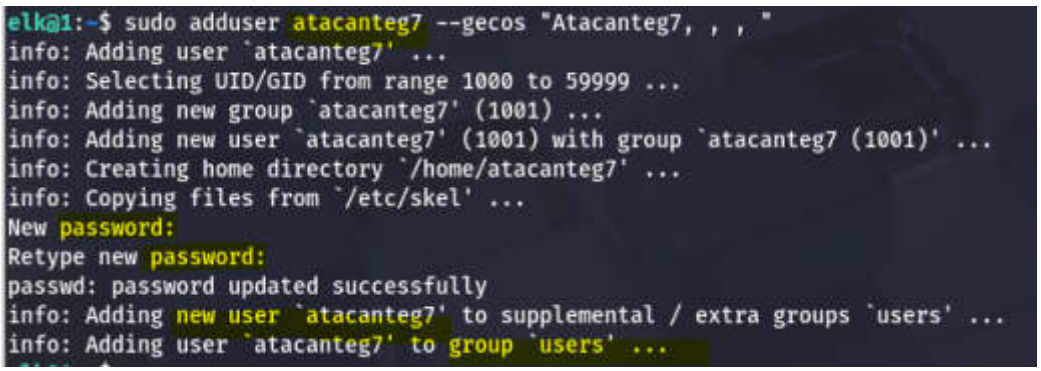


Creando usuario señuelo en SSH

```
sudo adduser atacanteg7 --gecos "Atacanteg7, , , "
```

Figura 110

Creación de usuario señuelo en SSH



Asegura autenticación por contraseña en SSH

```
sudo sed -i 's/^#\?PasswordAuthentication.*/PasswordAuthentication
yes/' /etc/ssh/sshd_config
sudo systemctl restart ssh
```

Figura 111

Reinicio del servicio SSH luego de crear el usuario señuelo

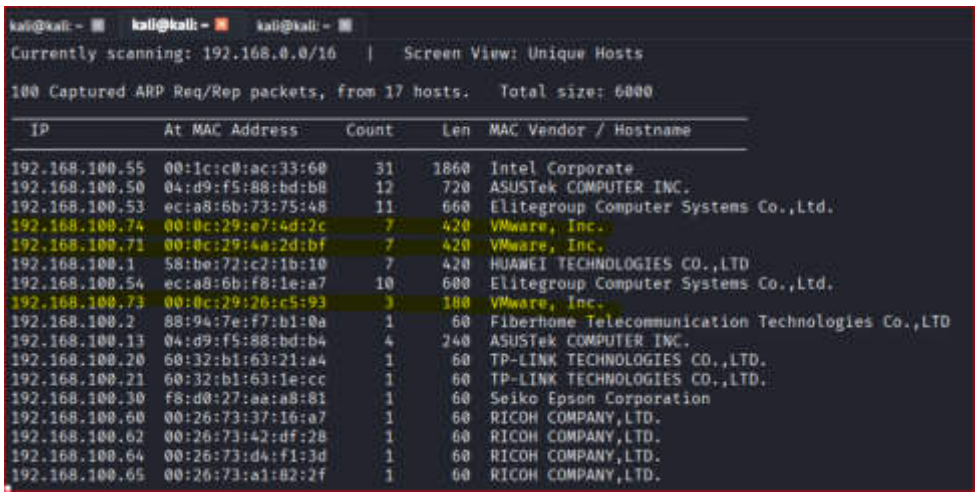
Reconocimiento desde Kali

Lanzamos netdiscover para descubrir las redes que tenemos

```
sudo netdiscover
```

Figura 112

Reconocimiento desde Kali con netdiscover



Lanzamos nmap para descubrir las redes que tenemos manera más agresiva

```
nmap -sn -PE -PS -PA -PU -PR \
10.0.0.0/8 \
172.16.0.0/12 \
192.168.0.0/16 \ 192.168.10.0/24 \ 192.168.100.0/24
```

Las redes que nos interesan son 192.168.100.74, 192.168.100.73, 192.168.100.71 las cuales vamos a realizar los ataques

Guardamos la IP del servidor ELK (192.168.100.74) realizamos este paso por cada red:

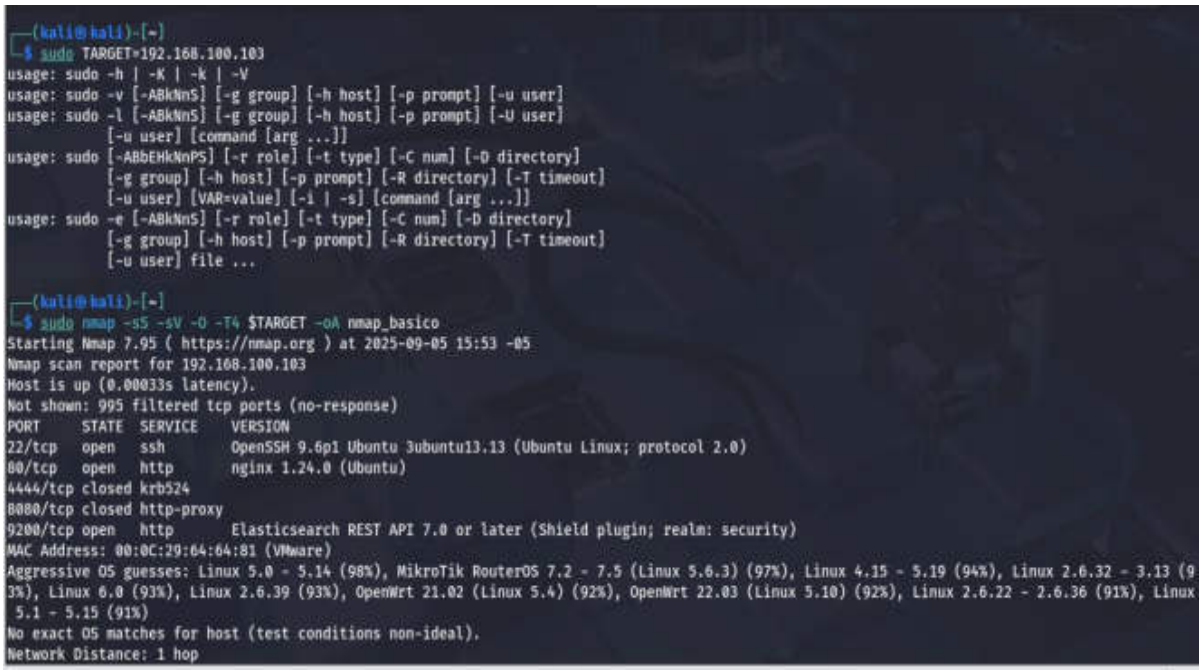
```
sudo TARGET=192.168.100.71 / sudo TARGET=192.168.100.73 / sudo TARGET=192.168.100.74
```

Escaneamos los puertos + versión (ruido moderado)

```
nmap -sS -sV -O -T4 $TARGET -oA nmap_basico
```

Figura 113

Escaneo de puertos y versión con ruido moderado

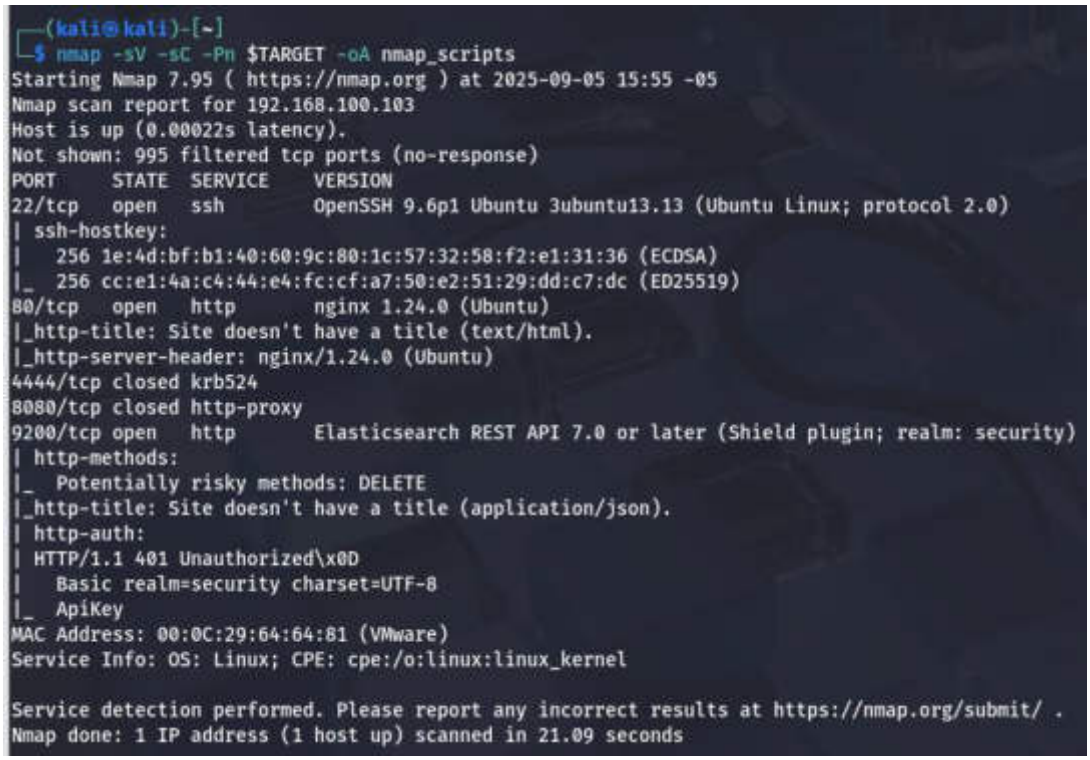


Escaneo más profundo de servicios abiertos detectados

```
nmap -sV -sC -Pn $TARGET -oA nmap_scripts
```

Figura 114

Escaneo profundo de servicios abiertos detectados



Información general

Host: 192.168.100.71

Sistema operativo: Linux (Ubuntu 24.04 con kernel 6.8.x, detectado por fingerprint de servicios).

Fabricante NIC: VMware → la máquina está virtualizada.

Puertos y servicios encontrados

`22/tcp → SSH (abierto)`

Servicio: OpenSSH 9.6p1 (Ubuntu) / Permite acceso remoto seguro por consola.

Fingerprints de hostkey (ECDSA y ED25519) identificados.

```
80/tcp → HTTP (abierto)
```

Servicio: Nginx 1.24.0 (Ubuntu).

Página web sin título (probablemente un sitio por defecto o servicio básico).

```
4444/tcp → Cerrado (krb524)
```

Puerto accesible pero cerrado.

```
8080/tcp → Cerrado (http-proxy)
```

También detectado como cerrado.

```
9200/tcp → HTTP (abierto)
```

Servicio: Elasticsearch 7.x o superior con Shield plugin (seguridad habilitada).

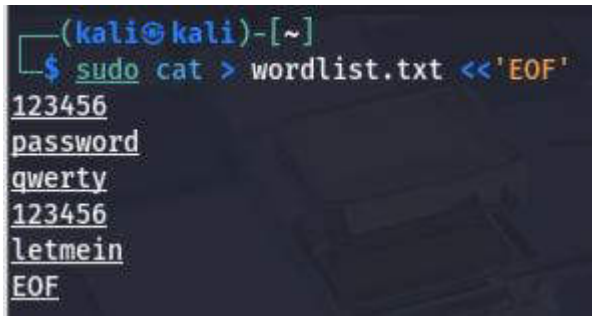
Autenticación: Basic realm=security + soporte para ApiKey.

Métodos HTTP disponibles incluyen DELETE (riesgo si no está bien protegido).

Fuerza bruta controlada a SSH (para poblar auth.log)

Figura 115

Ingreso de palabras en wordlist.txt



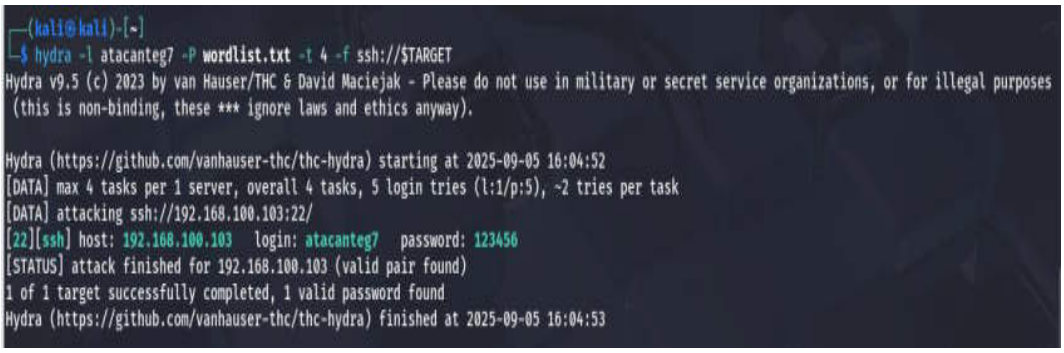

```
sudo cat > wordlist.txt <<’EOF’
123456
Password
Qwuery
12345567
Letmein
Elk
Siem
elkstackG7
EOF
```

Lanzamos HYDRA con baja concurrencia:

```
hydra -l atacanteg7 -P wordlist.txt -t 4 -f ssh://$TARGET
```

Figura 116

Lanzamiento de HYDRA con baja concurrencia



Resumen

El ataque de fuerza bruta con Hydra contra el servicio SSH (22) en 192.168.100.71 permitió descubrir credenciales válidas: usuario atacanteg7 con contraseña 123456, evidenciando una configuración insegura con contraseñas débiles (usuario creado anteriormente).

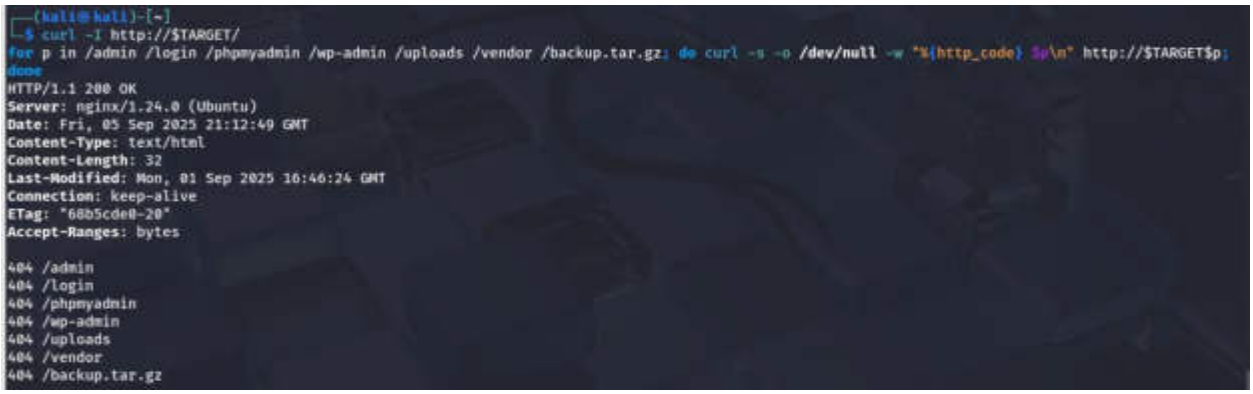
Realizamos otros tipos de ataques desde KALI

Barridos 404/403 y user-agents típicos

```
curl -I http://$TARGET/
for p in /admin /login /phpmyadmin /wp-admin /uploads /vendor
/backup.tar.gz; do curl -s -o /dev/null -w "%{http_code} $p\n"
http://$TARGET$p; done
```

Figura 117

Barridos 404/403



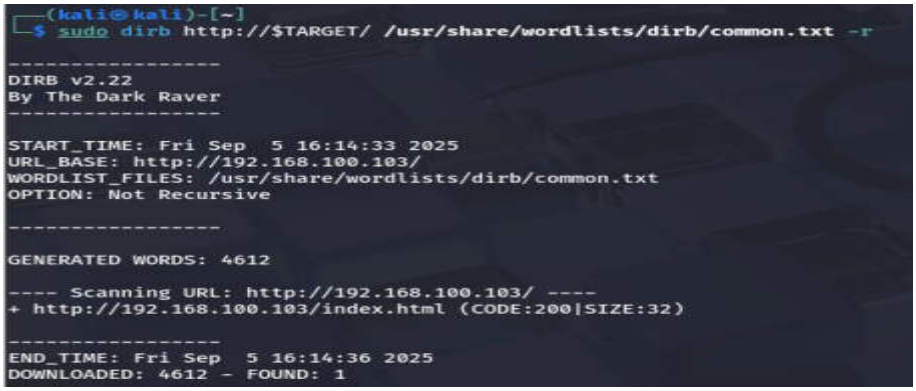
Resumen:

El análisis con curl sobre las redes 192.168.100.71/ 73/ 74/ XX confirma que el servidor web (nginx 1.24.0 en Ubuntu) expone únicamente el archivo estático index.html, mientras que rutas sensibles como /admin, /login o /phpmyadmin devuelven 404 Not Found, indicando un entorno con contenido mínimo y sin paneles administrativos accesibles.

```
Dirb/ffuf (NO destructivo)
sudo dirb http://$TARGET/ /usr/share/wordlists/dirb/common.txt -r
```

Figura 118

Configuración de Dirb/ffuf



Resumen:

El escaneo de directorios web con Dirb sobre el host 192.168.100.103 detectó únicamente la página /index.html (código 200, tamaño 32B), lo que indica un servidor web con contenido mínimo o en configuración por defecto.

Creamos un usuario en el SIEM

```
sudo curl -u atacanteg7:123456 -X POST
"http://192.168.100.71:9200/_security/user/nuevo_usuario" -H "Content-Type:
application/json" -d '{
  "password" : "ContraseñaSegura123!",
  "roles" : [ "kibana_user" ],
  "full_name" : "Nuevo Usuario Kibana",
  "email" : "nuevo@dominio.com"
}'
```

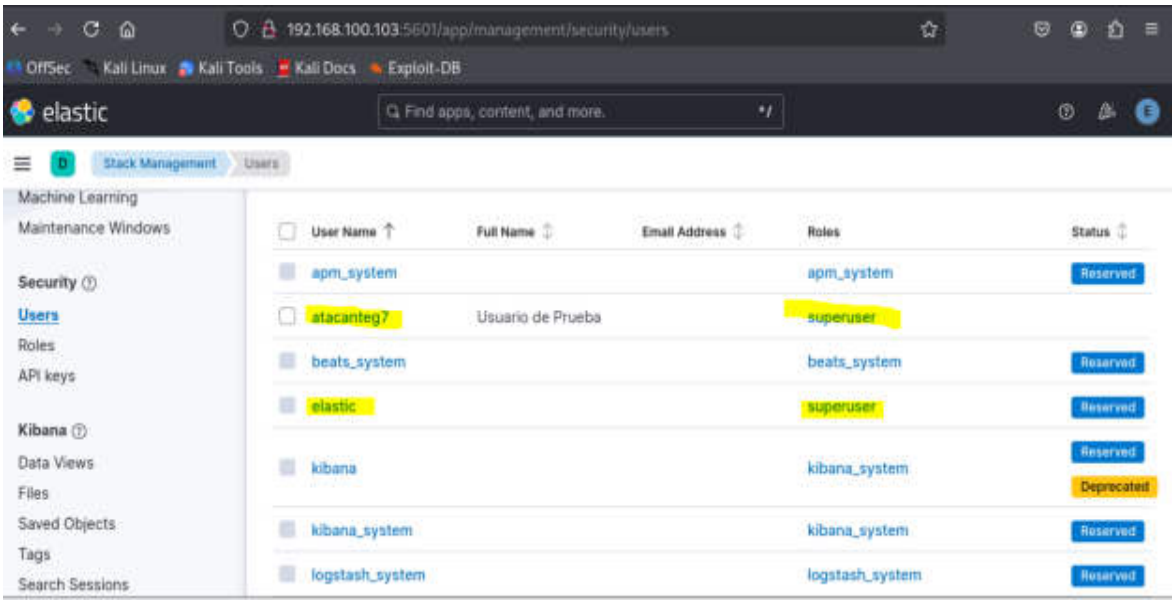
Figura 119

Creación de nuevo usuario en SIEM

Verificamos que el usuario se creó correctamente en KIBANA

Figura 120

Verificación de creación de nuevo usuario en Kibana



Con el usuario creado y con los permisos de SUPERUSER podemos realizar cualquier tipo de acción para ello vamos a empezar con:

Consultar todos los índices:

```
curl -u atacanteg7:123456 -X GET
http://192.168.100.103:9200/_cat/indices?v
```

Figura 121

Consulta de índices en nuevo usuario

Creamos un índice de prueba

```
curl -u atacanteg7:123456 -X PUT
http://192.168.100.103:9200/indice_prueba
```

Figura 122

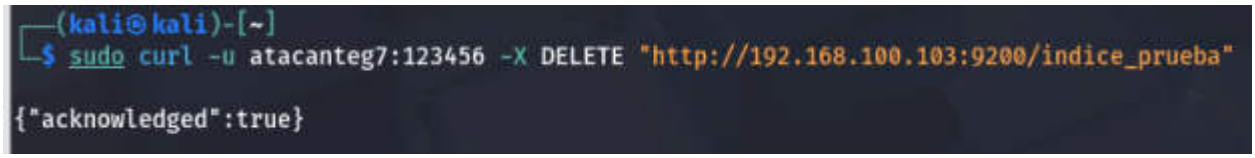
Creación de índice de prueba

Borramos el índice de prueba

```
curl -u atacanteg7:123456 -X DELETE
"http://192.168.100.103:9200/indice_prueba"
```

Figura 123

Eliminado de índice de prueba

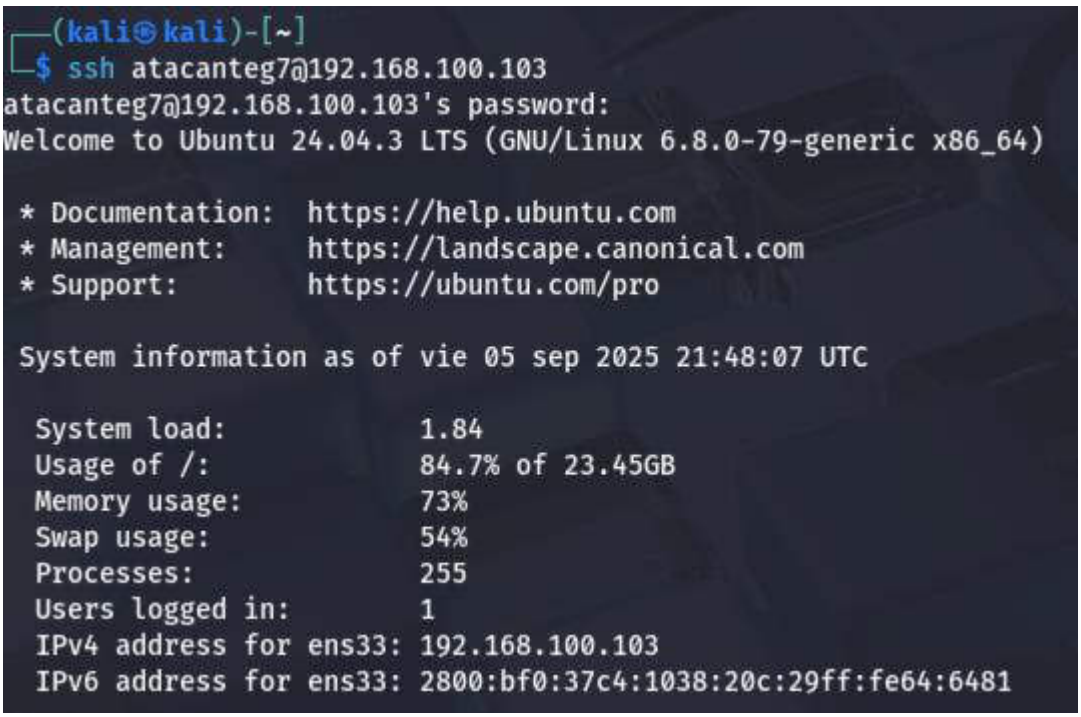


Detenemos Kibana / Elasticsearch

```
ssh atacanteg7@192.168.100.103
```

Figura 124

Detención de los servicios de Kibana y Elasticsearch



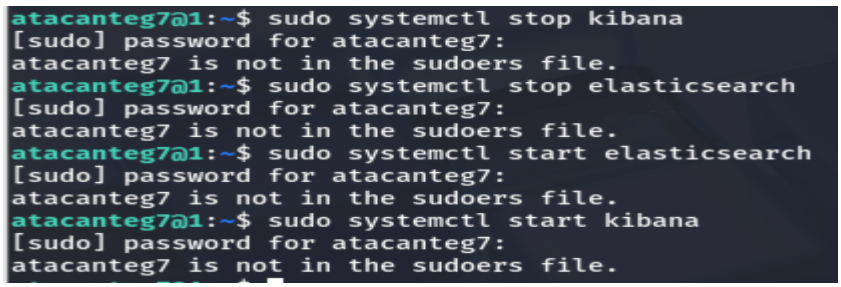
Una vez dentro del servidor ejecutamos

```
sudo systemctl stop kibana

sudo systemctl stop elasticsearch
```

```
sudo systemctl start elasticsearch

sudo systemctl start kibana
```

Figura 125

Reincio de servicio de Kibana



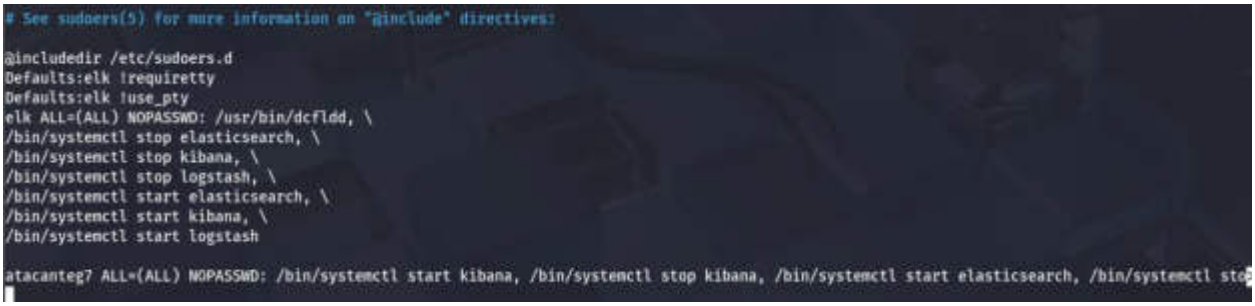
Como se observa el superusuario creado solo nos permite hacer cualquier cambio dentro del entorno del SIEM para poder hacerlo desde el servidor de Ubuntu hay que configurar sudoers, para ello ingresamos con la cuenta elk

Agregamos esta línea de comando

```
atacanteg7 ALL=(ALL) NOPASSWD: /bin/systemctl start kibana,
/bin/systemctl stop kibana, /bin/systemctl start elasticsearch,
/bin/systemctl stop elasticsearch
```

Figura 126

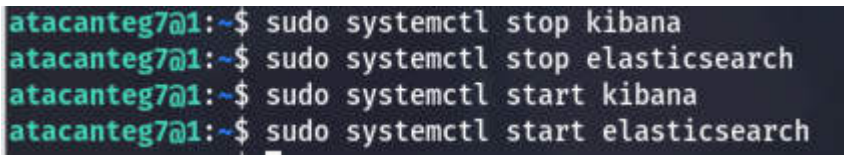
Configuración de sudoers en ELK



Guardamos y volvemos a ejecutar los comandos

Figura 127

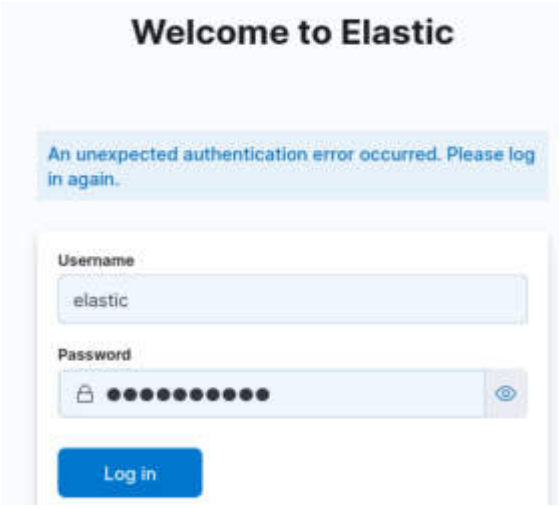
Nuevo reinicio de Kibana y Elasticsearch



Como observamos con el usuario creado desde HYDRA ya podemos hacer cualquier cambio dentro del servidor y dentro del SIEM.

Figura 128

Nueva verificación del SIEM



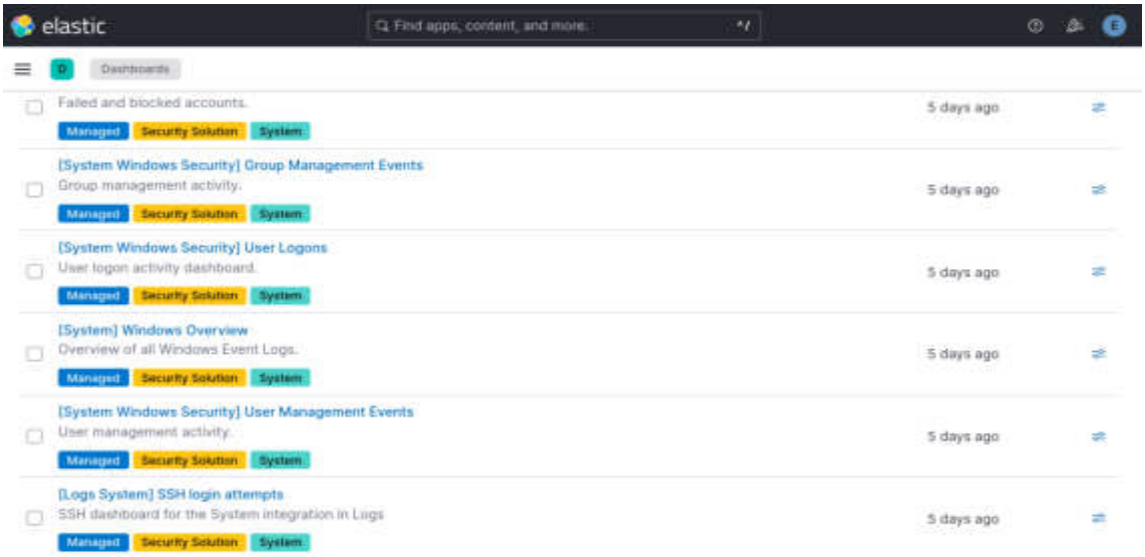
Análisis Forense

Iniciamos el análisis forense verificando físicamente los logs en el SIEM para posteriormente sacar imágenes forenses de la RAM de la partición que nos interesa.

En KIBANA revisamos el estado del equipo los logs que ya tenemos en nuestro SIEM

Figura 129

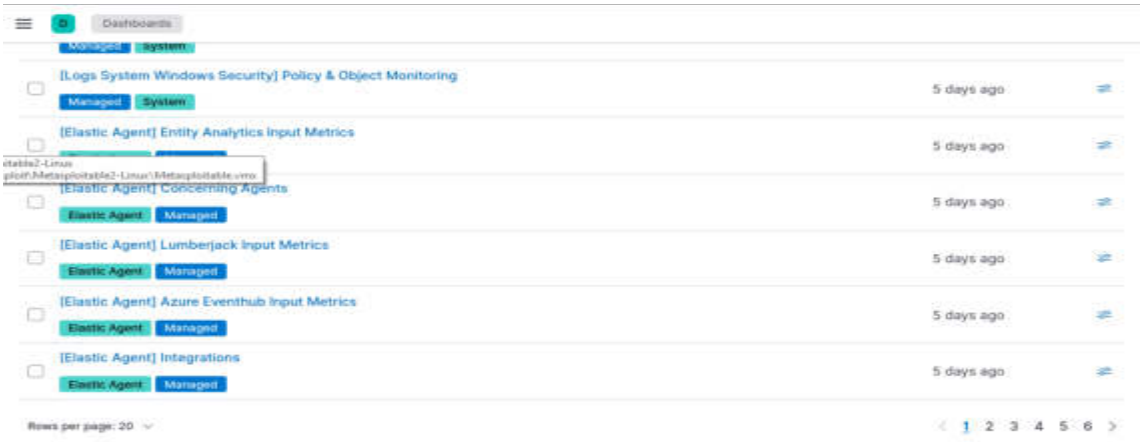
Revisión de estado del equipo en Elastic



Tenemos diversos tipos de logs, entre ellos se puede evidenciar ingresos fallidos, incidentes de seguridades, incidentes por ssh, Grupos de usuarios etc.

Figura 130

Constatación de distintos tipos de logs en Elastic

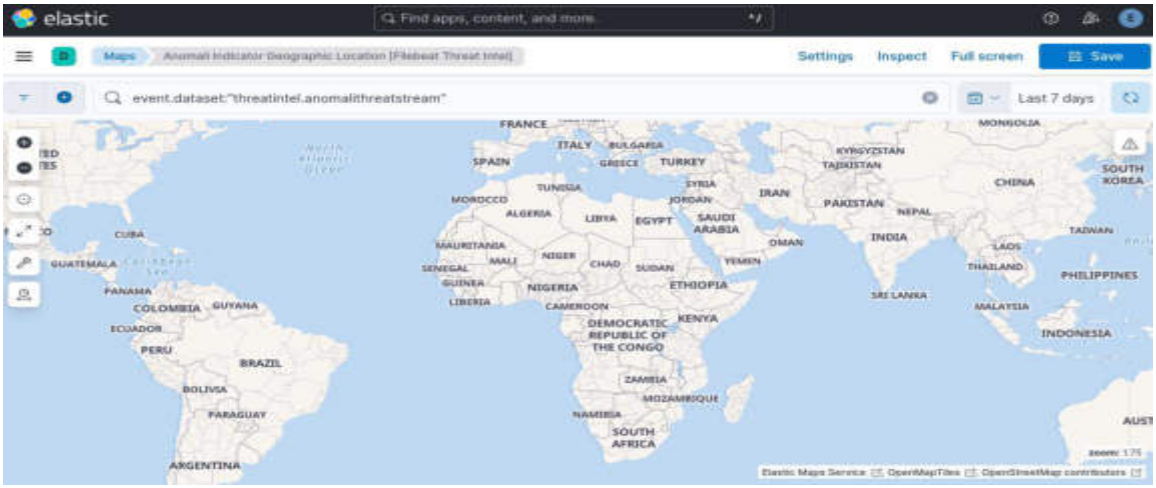


También podemos evidenciar registros exitosos.

Adicional a esto también podemos verificar los accesos por medio de mapas, en este apartado podemos determinar de donde provienen las peticiones al SIEM.

Figura 131

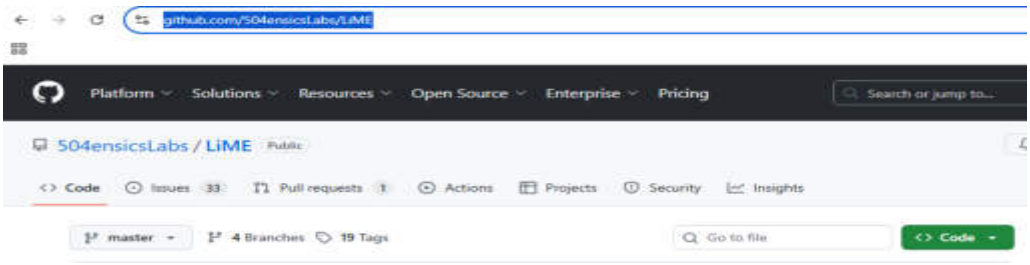
Verificación de accesos por medio de mapas



Para realizar el volcado de la memoria en tiempo real con el equipo en ejecución procedemos a instalar LiME, nos descargamos de:

Figura 132

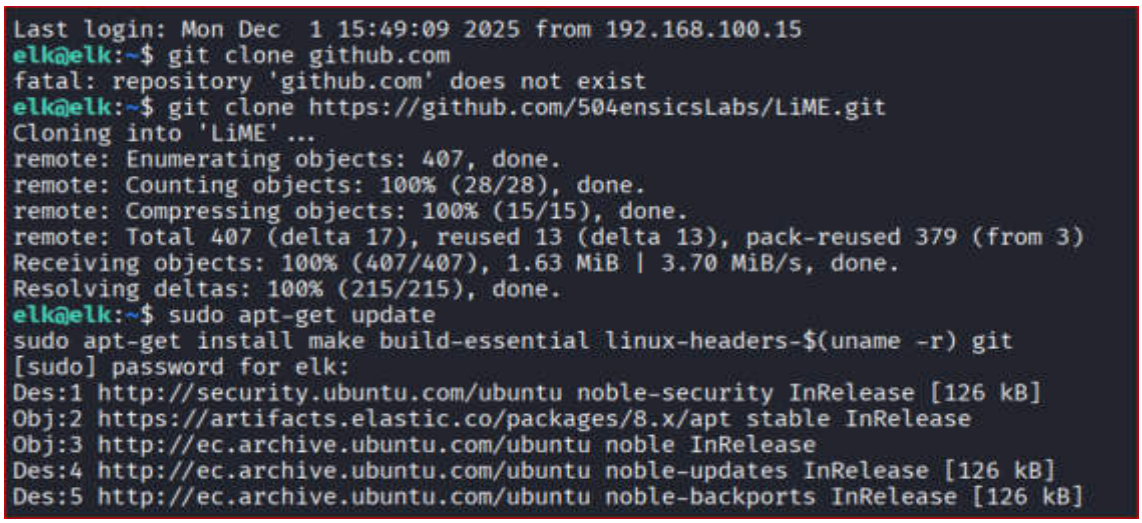
Descarga de LiME desde GitHub



Desde el repositorio nos descargamos los binarios de LiME

Figura 133

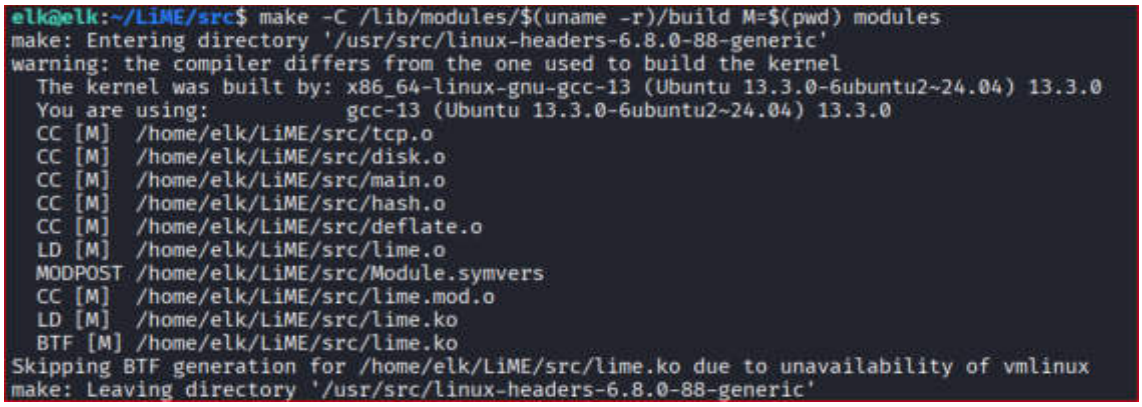
Descarga de binarios de LiME



Reconstruimos el binario para poder instalar

Figura 134

Reconstrucción del binario para instalación de LiME



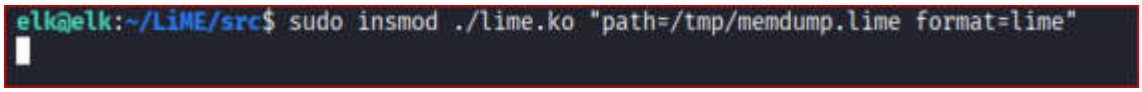
Volcar localmente y después enviarlo por SCP (seguro)

Volcar la RAM en ELK:

```
sudo insmod ./lime.ko "path=/tmp/memdump.lime format=lime"
```

Figura 135

Volcado de RAM en ELK



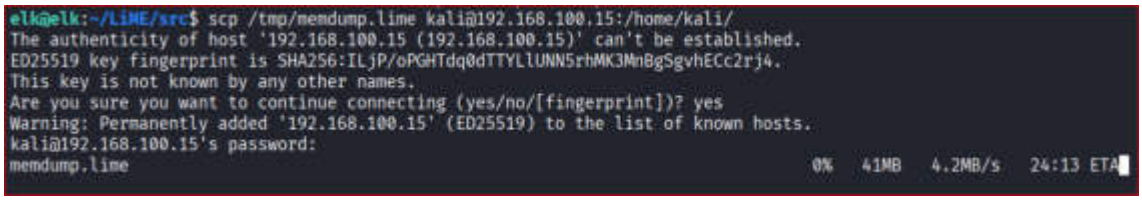
Enviar el archivo a Kali:

Desde el SIEM ELK ejecutamos:

```
scp /tmp/memdump.lime kali@192.168.100.15:/home/kali/
```

Figura 136

Envío de archivo a Kali



Avance de la transferencia del archivo .lime

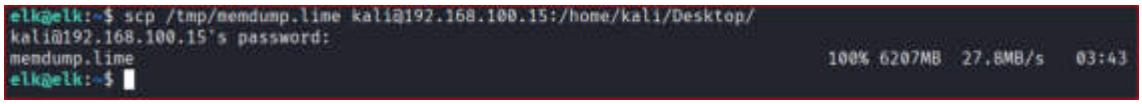
Figura 137

Transferencia de archivo .lime

Transferencia terminada con éxito

Figura 138

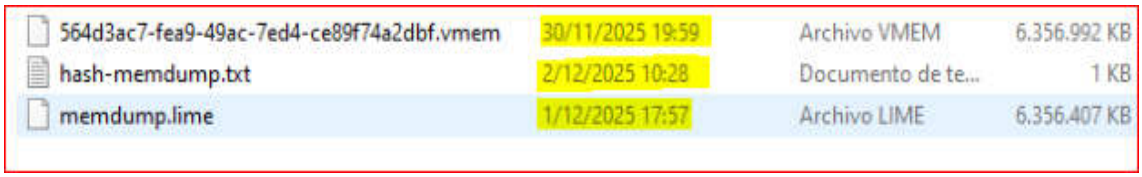
Transferencia terminada con éxito



Fecha de la imagen extraída, también se puede utilizar la RAM de la máquina virtual pero en nuestro caso estamos simulando un escenario real

Figura 139

Fecha de la imagen extraída



Al analizar el archivo extraído verificamos que no obtuvimos mucha información por lo que se procede a extraer la imagen forense por red utilizando el comando **dcfldd**

"**dcfldd** es una versión mejorada del comando dd de Linux diseñada para el análisis forense digital, que ofrece funciones como "hashing on-the-fly" (cálculo de hashes simultáneo a la transferencia) para asegurar la integridad de los datos, salidas múltiples y a distintos destinos, y la capacidad de verificar la imagen o el borrado de un disco. También proporciona información de progreso detallada y una mayor eficiencia de transferencia al utilizar un tamaño de bloque predeterminado más grande que dd.

Características clave de **dcfldd**:

```
Hashing on-the-fly:
```

Calcula valores de hash (como MD5 o SHA) de los datos mientras se copian o se procesan, garantizando que la imagen o el borrado no se alteren.

Verificación de imagen/borrado:

Permite verificar que una unidad de destino sea un duplicado exacto (bit a bit) de un archivo o patrón de entrada.

Salidas múltiples:

Puede enviar la salida a varios archivos o discos simultáneamente.

Salidas divididas (Split Output):

Divide la salida en múltiples archivos con más opciones de configuración que el comando split.

Salida de estado y registro:

Proporciona una salida de progreso que muestra la cantidad de datos transferidos y el tiempo estimado restante, con la posibilidad de dirigir los logs y la salida a otros comandos.

Borrado de discos:

Facilita el borrado rápido y seguro de discos utilizando un patrón conocido, si se especifica.

Mayor eficiencia:

Utiliza un tamaño de bloque de 32 KiB por defecto, mucho más grande que el de 512 bytes de dd, lo que mejora significativamente el rendimiento.

Propósito:

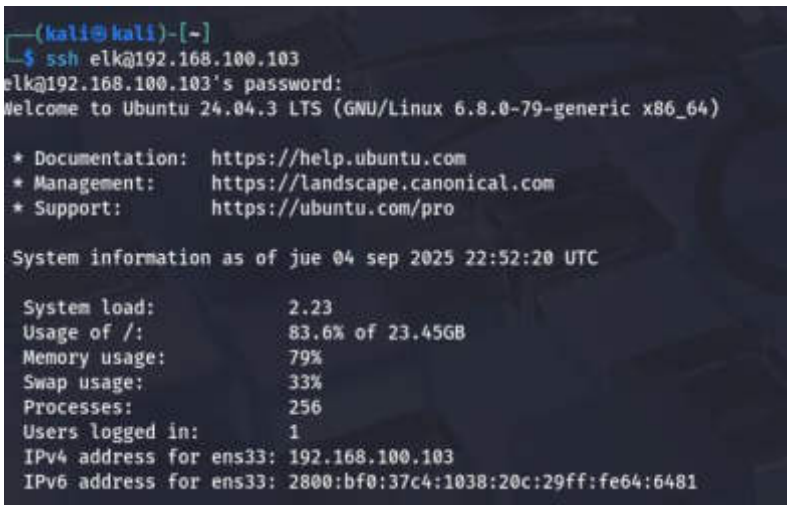
dcfldd es una herramienta esencial para los profesionales de informática forense y seguridad, ya que su diseño se enfoca en la confiabilidad y en la integridad de los datos durante la adquisición de imágenes de discos duros y otras operaciones de copia.”

Realizamos la extracción de la imagen forense solo de la partición donde se encuentran todos los logs de ELKSTACK ya que si obtenemos la imagen forense de todo el disco es demasiado grande por ello optamos por extraer la imagen solo de la partición que nos interesa para su análisis

Nos conectamos al servidor vía ssh

Figura 140

Conexión al servidor SSH desde Kali



Con el comando sudo visudo procedemos a configurar nuestro SIEM para que no nos pida la contraseña al momento de realizar la imagen forense

Ingresamos al archivo de configuración

Figura 141

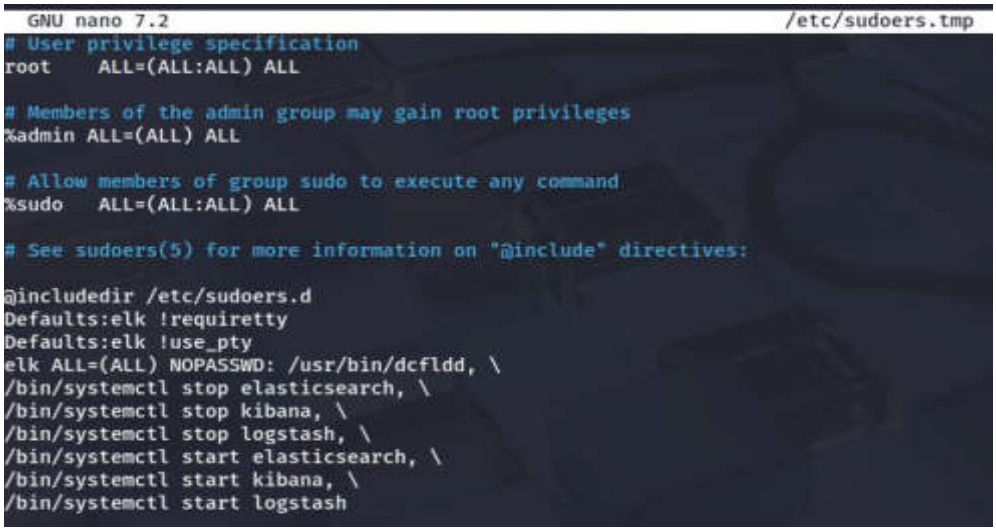
Ingreso al archivo de configuración en Kali

Agregamos las siguientes líneas de comando:

```
Defaults:elk !requiretty
Defaults:elk !use_pty
elk ALL=(ALL) NOPASSWD: /usr/bin/dcfldd, \
/bin/systemctl stop elasticsearch, \
/bin/systemctl stop kibana, \
/bin/systemctl stop logstash, \
/bin/systemctl start elasticsearch, \
/bin/systemctl start kibana, \
/bin/systemctl start logstash
```

Figura 142

Configuraciones del archivo en Kali

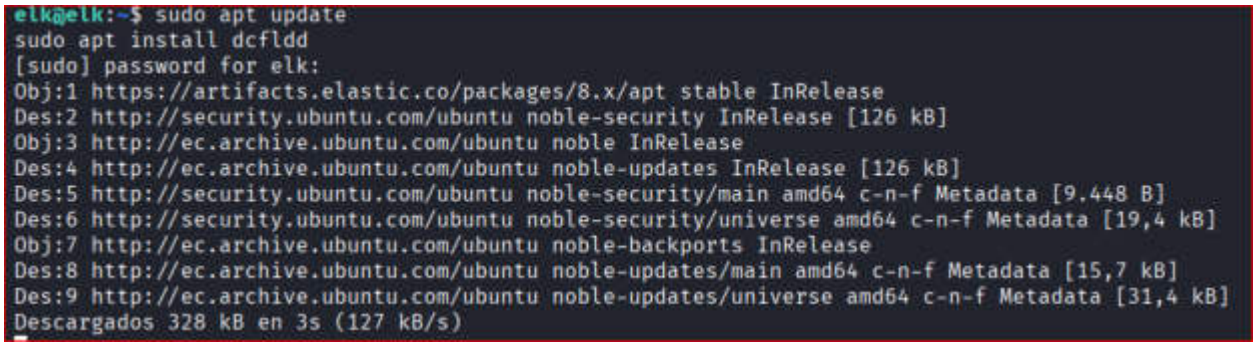


Guardamos y desde el host de Kali procedemos a crear la imagen forense

Primero procedemos a instalar dcfldd en el SIEM y en Kali para poder sacar la imagen forense en caliente.

Figura 143

Instalación de dcfldd en el SIEM



Una vez instalado ejecutamos el siguiente comando desde el host de kali

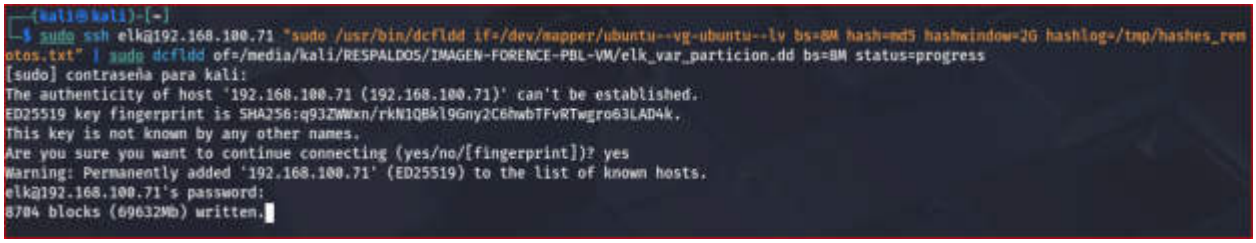
```
sudo ssh elk@192.168.100.71 "sudo /usr/bin/dcfldd
if=/dev/mapper/ubuntu--vg-ubuntu--lv bs=8M hash=md5 hashwindow=2G
hashlog=/tmp/hashes_remotos.txt" | sudo dcfldd
```

```
of=/media/kali/RESPALDOS/IMAGEN-FORENCE-PBL-VM/elk_var_particion.dd bs=8M
status=progress
```

Este comando nos permite sacar la imagen a un disco externo debido a que no hay espacio en el host de KALI, tomando en cuenta la integridad de la imagen y la línea de tiempo

Figura 144

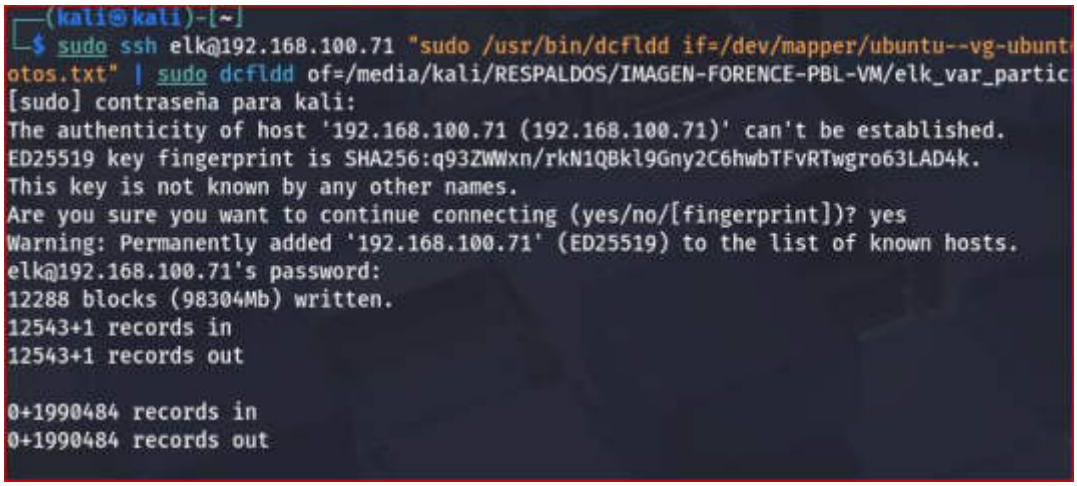
Extracción de la imagen a un disco externo



Verificamos que se crea la imagen elk_var_particion.dd y pesa 98.3GB

Figura 145

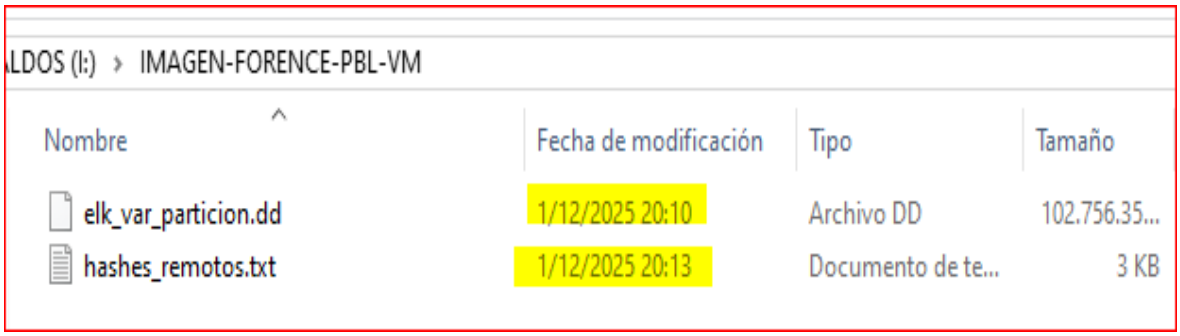
Verificación de creación de la imagen elk_var_particion.dd



Fecha de creación de la imagen Forense

Figura 146

Fecha de creación de imagen forense

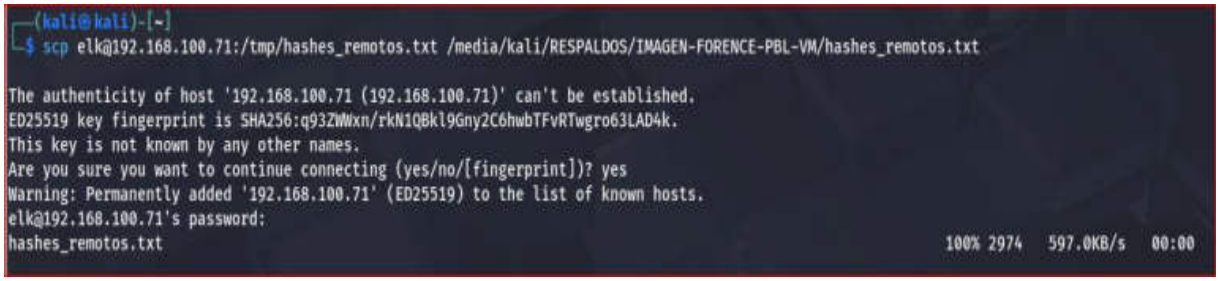


Este comando dcfldd nos permite extraer los SHA256 al mismo tiempo, procedemos a realizar la copia del archivo que contiene los HASH y lo pasamos al equipo de Kali.

```
scp elk@192.168.100.71:/tmp/hashes_remotos.txt
/media/kali/RESPALDOS/IMAGEN-FORENCE-PBL-VM/hashes_remotos.txt
```

Figura 147

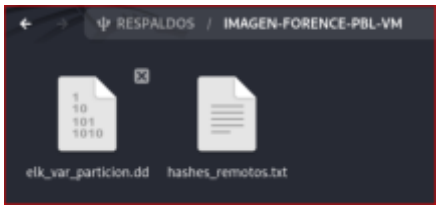
Extracción de hashes SHA256 con dcfldd



Verificamos que en nuestro equipo de KALI tengamos los dos archivos credos, la imagen forense vía red y los HASH extraídos de la imagen forense.

Figura 148

Verificación de archivos creados en VM Kali



Los HASHES los va creando conforme va extrayendo la imagen los guarda von el nombre hashes_remotos.txt y estos son:

```
0 - 2147483648: ab626758400d367bb5e8fdb731c77130
2147483648 - 4294967296: 313c4e5a9e5ddf26abbce51c5fd30635
4294967296 - 6442450944: 18fe3ff5296ea23e083ae3471844114e
6442450944 - 8589934592: 85746385429e83bcce6e98cec9677519
8589934592 - 10737418240: 7d8fcc2495c3f4a8d30fb679ce1e364f
10737418240 - 12884901888: 322359849dfa9f06a4eda4b0e8be336d
12884901888 - 15032385536: 0c9c1dd470f1bee48cc3e0ada2527e75
15032385536 - 17179869184: 1ec9b7f60eb5e153a7a3aa8168b76cd9
17179869184 - 19327352832: 9a44ab5ba1482ad8770ad1141d0a2740
19327352832 - 21474836480: ab4185bc2f62b42e85bfd030f0860f1e
21474836480 - 23622320128: dcdc5dfd3ee53617f639e67f2caf946b
23622320128 - 25769803776: 7ff9eca9b727f9f6cc87b04b6f094553
25769803776 - 27917287424: 06da341c8c4bc687a3d63924af411b54
27917287424 - 30064771072: 7f5facd04288f3d243e92f8e87fdd459
30064771072 - 32212254720: 65429c2f281af12724c3720118f85b12
32212254720 - 34359738368: 8aba90a395af692c7006b6ff8fbdd800
34359738368 - 36507222016: 91833554f9f98420ebf1f9b391597e9d
36507222016 - 38654705664: 649d6761423d2b7ec2204f272f3e43a8
38654705664 - 40802189312: def389a5ef0e5288e1d3805581ede443
40802189312 - 42949672960: 9257aae8e2022243ea71b453c1e838ab
```

```
42949672960 - 45097156608: 37693acbdfabe56e3b53ce6d2a0c5c90

45097156608 - 47244640256: 1cc01695692e416587034695943c5a95

47244640256 - 49392123904: edcd96b9baf5e11395812e200b30022d

49392123904 - 51539607552: 3c135f84194118d5049d8e357054113d

51539607552 - 53687091200: 3542dc37abca85f48b59018ef7c09f82

53687091200 - 55834574848: 2c485f2ef5a9288a37917fc88bd2f0b6

55834574848 - 57982058496: 92d0c3ba8d80fd422875a715985bcbd7

57982058496 - 60129542144: 3962106bc22126d73457bfc081f8247a

60129542144 - 62277025792: a4286098bf8427ca58fd11940077cf9c

62277025792 - 64424509440: 0a1f1d4458341df6d16ce2688a09df98

64424509440 - 66571993088: cade8856e5368d80861f62a809fb9f22

66571993088 - 68719476736: b1227be917eb5030e981108b713ee607

68719476736 - 70866960384: 746258aeb3b6bf59900e330a50ddf8a3

70866960384 - 73014444032: f3f90a7f6219b5353d871dd27d70c180

73014444032 - 75161927680: 0d65a11d1daea9f25b457a166fa37a81

75161927680 - 77309411328: 19ead08afd8af05055a39bb7e776f63c

77309411328 - 79456894976: 22bbba9a6c6df8619c84dc45a61d8069

79456894976 - 81604378624: 25fb15be394d8280e4710b24494e00a6

81604378624 - 83751862272: 806b5cf2a5887d2af99152161d12ca5a

83751862272 - 85899345920: 8ba93b6f60e5b5c16a5c0b633778c70f

85899345920 - 88046829568: 61335e862df361dbac82b23b9dff3296

88046829568 - 90194313216: 58f55c1e685deb8cbf08f15ebeb77329

90194313216 - 92341796864: 02136c41bcb00a0ca56f37635bf86144

92341796864 - 94489280512: de1b27dbd1f2b526df5064237d9015eb

94489280512 - 96636764160: fba11ec593b988824679c16bd0e69d3c

96636764160 - 98784247808: a87e976d7548d66a79f8c11c5b8c7aa4

98784247808 - 100931731456: 67220e412a342dd33d6ccde40231dbe4

100931731456 - 103079215104: 6da3fe064c38b6a93639e7eeaf84a73c

103079215104 - 105222504448: 1cf0374bf36ffbcbb48849af5a2532d8


Total (md5): 9371ef96ba0c8783af7d2c0175802d33
```

Procedemos a realizar el análisis forense con las diversas herramientas a la imagen extraída.

4. Análisis Forense De La Imagen elk_var_particion.dd con Autopsy 4.22.1

Datos Generales de la Evidencia

Nombre del archivo analizado: elk_var_particion.dd

Hash MD5: `9371ef96ba0c8783af7d2c0175802d33`

Hash SHA256:
`96306eb972ad9d5411e775c6214e37e0607bd371f91d27dfdbf2aa4f0157e70b`

Fecha de adquisición: 02/12/2025

Herramienta utilizada: Autopsy

Analista responsable: grupo7

Como podemos observar el hash MD5 al ser procesado en el repositorio para su análisis con fecha 02-12-2025 es el mismo que se obtuvo junto con la imagen extraída, esto nos indica que la línea de tiempo, custodia y procesamiento no han sido corrompidas

Figura 149

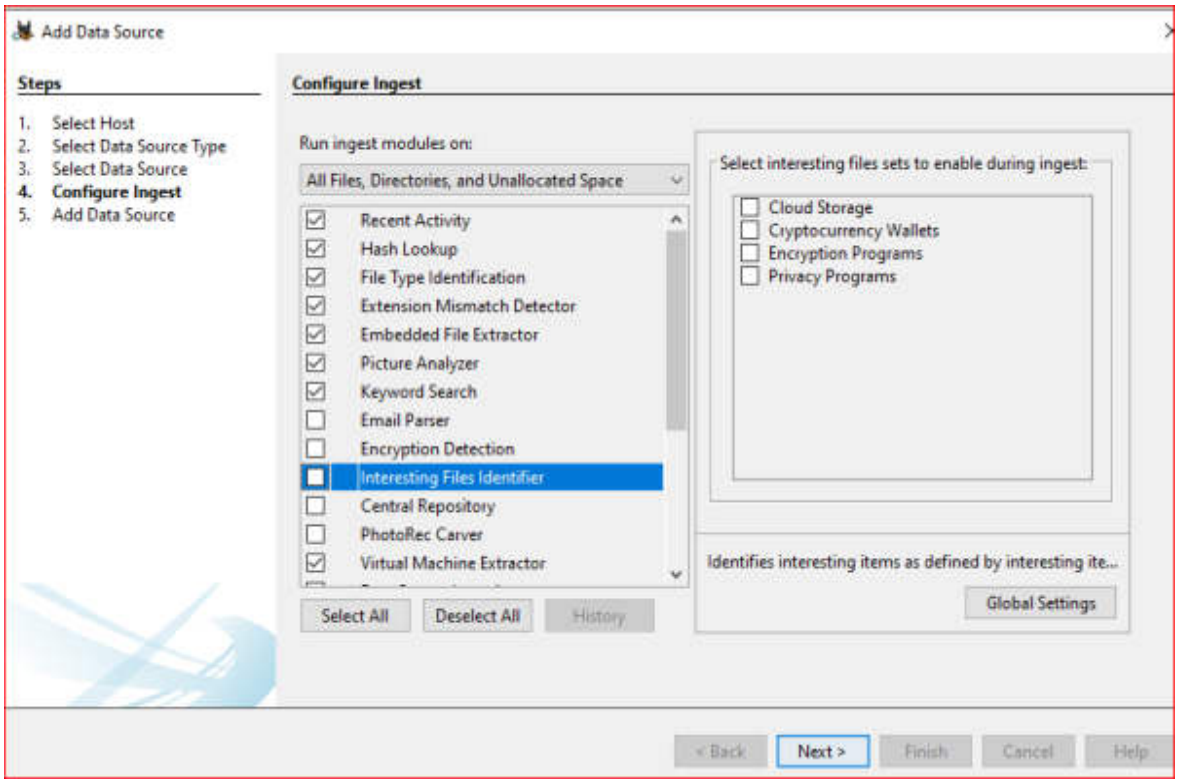
Hash MD5 de elk_var_particion.dd

```
PS I:\IMAGEN-FORENCE-PBL-VM> CertUtil -hashfile '.\elk_var_particion.dd' SHA256
SHA256 hash de .\elk_var_particion.dd:
96306eb972ad9d5411e775c6214e37e0607bd371f91d27dfdbf2aa4f0157e70b
CertUtil: -hashfile comando completado correctamente.
PS I:\IMAGEN-FORENCE-PBL-VM> CertUtil -hashfile '.\elk_var_particion.dd' MD5
MD5 hash de .\elk_var_particion.dd:
9371ef96ba0c8783af7d2c0175802d33
CertUtil: -hashfile comando completado correctamente.
PS I:\IMAGEN-FORENCE-PBL-VM>
```

Primero cargamos la imagen a AUTOPSY, esta carga va a depender del tamaño de la imagen forense, escogemos las opciones que necesitamos para el análisis.

Figura 150

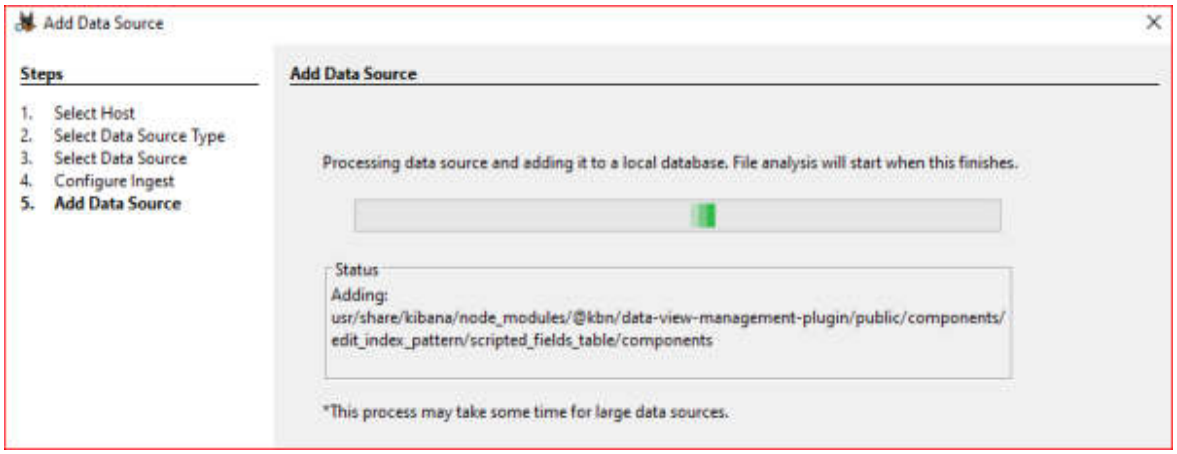
Carga de la imagen en Autopsy



Autopsy va cargando todas los módulos y programas que se encuentran instalados y ejecutados en la imagen forense.

Figura 151

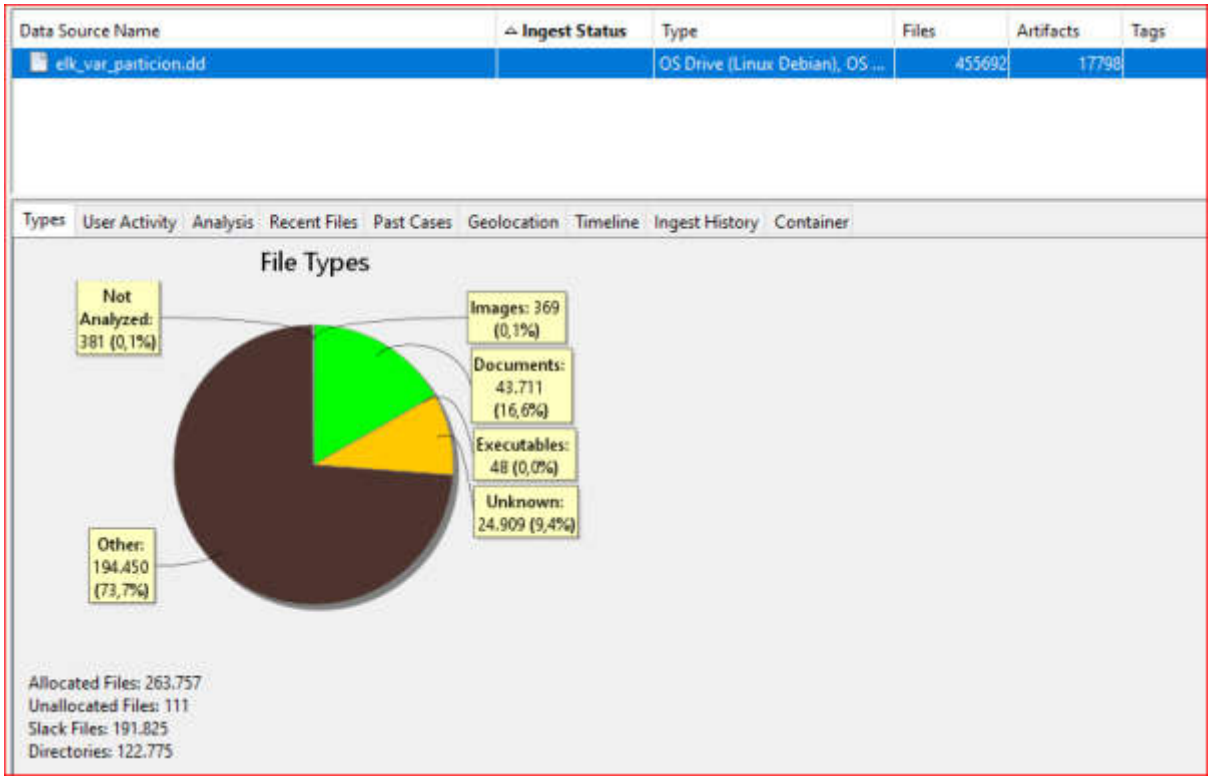
Se agrega la fuente en Autopsy



Sacamos una imagen del tamaño del disco a ser analizado

Figura 152

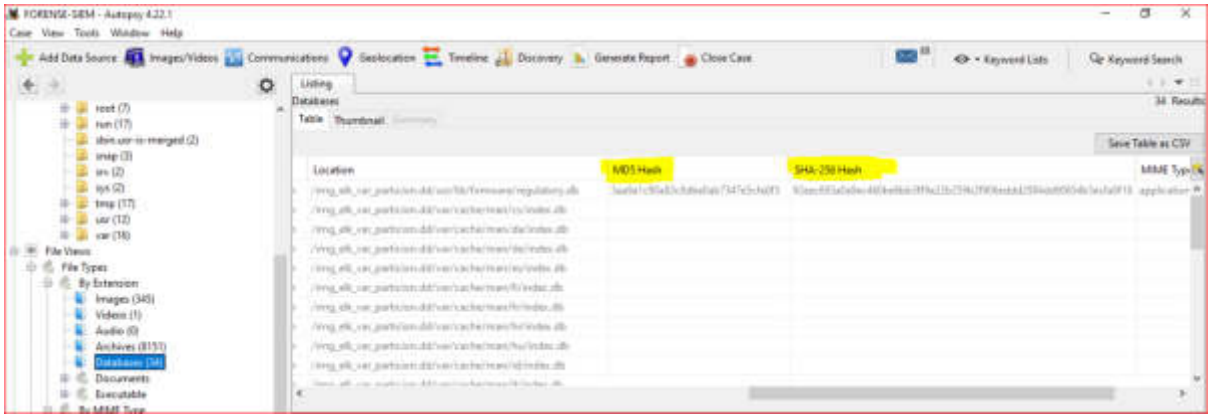
Extracción de imagen del disco a analizarse



Una vez cargada la imagen forense a nuestro Autopsy abrimos el archivo **Databases** y verificamos el HASH con el que fue creada la imagen y esta debe de coincidir con la extraída manualmente para nuestro registro.

Figura 153

Verificación de hashes en Autopsy

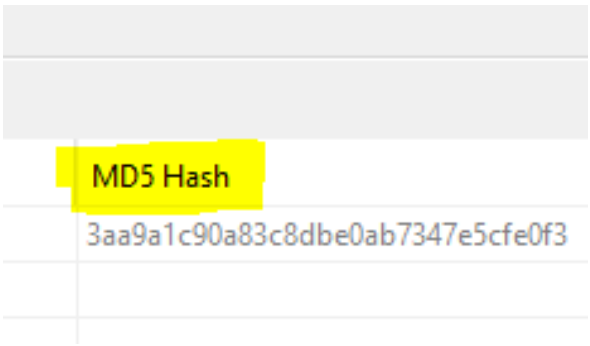


Los hashes

MD5

Figura 154

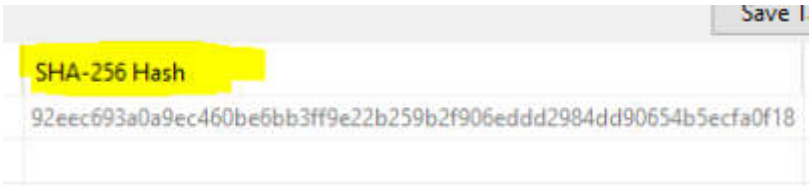
Verificación de hash MD5 en Autopsy



SHA256

Figura 155

Verificación de hash SHA256 en Autopsy



Línea de tiempo del análisis ejecutado

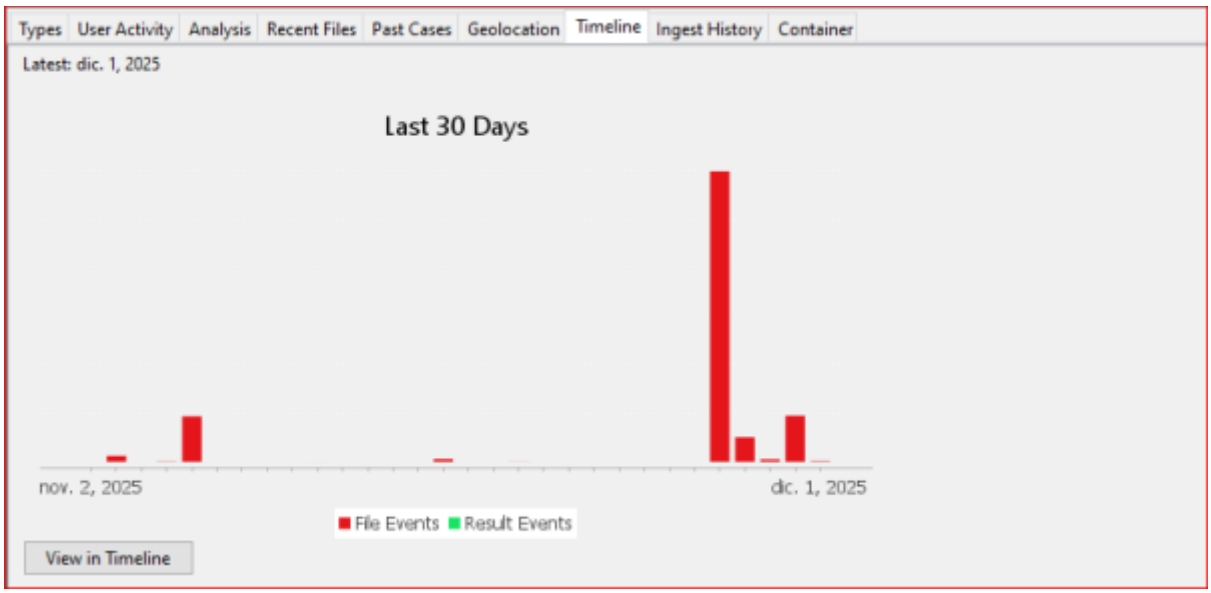
Figura 156

Línea de tiempo de análisis en Autopsy

Línea de tiempo de acuerdo con la fecha establecida

Figura 157

Línea de tiempo según la fecha establecida en Autopsy



Extraemos una Imagen Forense visible la cual nos indica:

Su Estructura de archivos y metadatos

A la izquierda se ve una jerarquía típica del sistema de archivos y categorías de análisis que se detallan:

Deleted Files Nos indica que hay archivos eliminados

MB File Size Nos indica el tamaños de archivos en megabytes

Data Artifacts Nos indica los artefactos de datos, incluyendo metadata

Analysis Results Nos da el resultado del análisis como metadatos EXIF, sospechas de contenido de usuario, y extensiones con errores)

Esto indica que el sistema está extrayendo y categorizando información relevante de la imagen **elk_var_particion.dd**

Análisis del archivo vxlan.jpg:

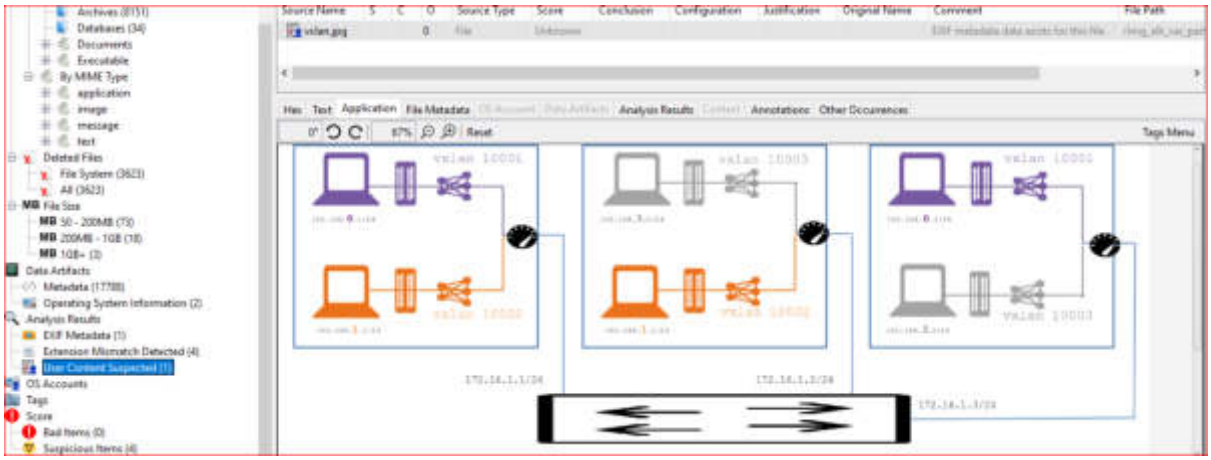
En el panel derecho, en la parte inferior tenemos "**Analysis Results**", esto nos muestra el análisis de un archivo llamado vxlan.jpg.

Se puede visualizar la información técnica, como la existencia de metadatos EXIF, que suelen contener datos sobre cuándo y cómo se tomó la imagen.

También está resaltado un posible contenido sospechoso ("**User Content Suspected**").

Figura 158

Detección de posible contenido sospechoso en Autopsy



Análisis Forense a /img_elk_var_particion.dd/var/log/apt

Se encontró los comandos ejecutados:

apt-get install make build-essential linux-headers-6.8.0-88-generic git

Requested-By: elk (1000)

Esto significa que el usuario elk (UID 1000) ejecutó:

Make, build-essential, linux-headers-X, git

Esto es típico de los que quieren compilar algún tipo de software, y compilar algo en un sistema en producción puede ser normal o puede ser muy sospechoso dependiendo del contexto.

¿ESTO ES MALICIOSO?

En este caso diremos que por sí solo NO es una prueba de ataque, PERO es un comportamiento característico de malware o rootkits, porque muchos atacantes instalan exactamente estos paquetes: build-essential, make, gcc, linux-headers, exclusivamente para poder:

- ✓ compilar un rootkit
- ✓ compilar un keylogger
- ✓ compilar un módulo del kernel
- ✓ compilar un backdoor persistente

Como análisis se podría determinar que si el sistema no era usado para desarrollo, esto es ALTAMENTE sospechoso.

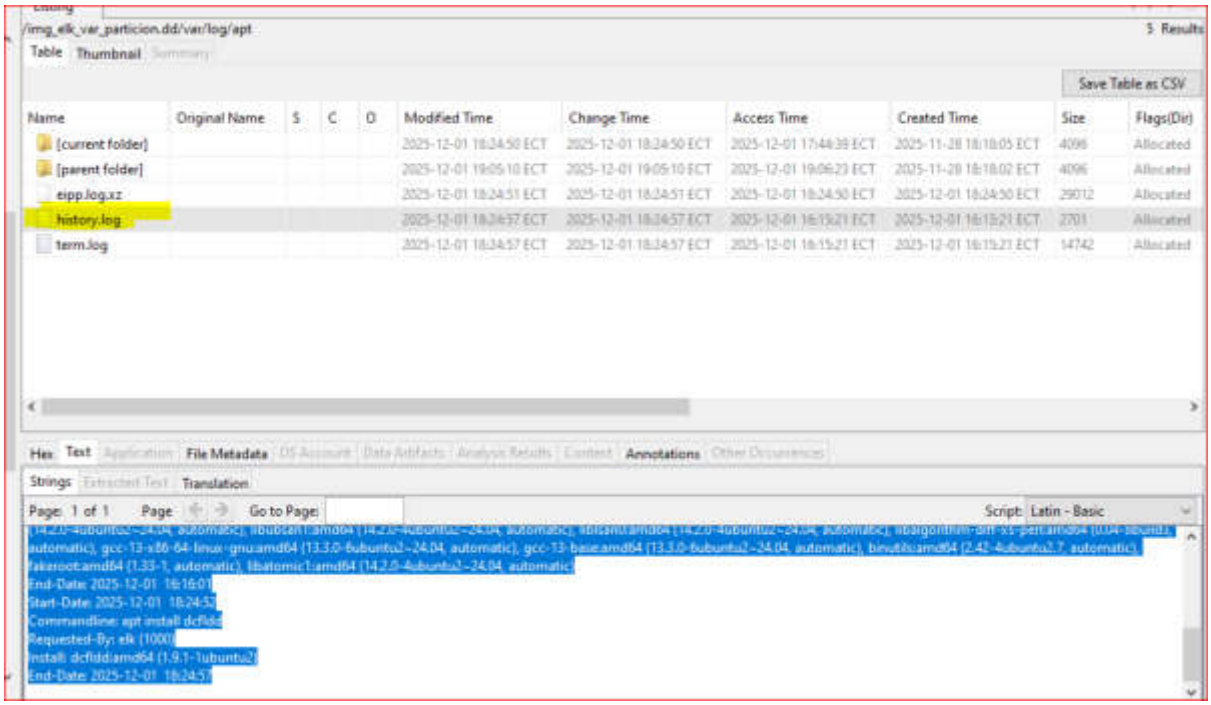
Otro evento relevante se tiene en:

- ✓ Start-Date: 2025-12-01 18:24:52
- ✓ Commandline: apt install dcfldd
- ✓ Requested-By: elk (1000)
- ✓ Install: dcfldd:amd64 (1.9.1-1ubuntu2)
- ✓ dcfldd: Es una herramienta forense

Esta es una versión mejorada de dd creada para imágenes forenses.

Figura 159

Versión mejorada del dd para imágenes forenses



Como evidencia se encontró el siguiente archivo MB 200BM - 1GB(18) se abrió _133cfs

Información adicional que aparece

192.168.100.15 / 192.168.100.71 es las direcciones IP de los hosts al que enviaron logs.

atacanteg7 es probablemente el nombre de usuario o de algún intentando de autenticarse.

systemd-logind, cron, (sd-pam) son procesos del sistema que generaron eventos.

filebeat, beats_input_codec_plain_applied estos registros nos indican que los logs fueron enviados a Elasticsearch mediante Filebeat.

Números y códigos largos

```
aafc8d365c904676b410af61cbf2507c
d008c31b-53ce-4b6c-ab1c-90250a450f36
172211eec75c
```

Son UUIDs, IDs de eventos o IDs efímeros (ephemeral_id) generados por Elasticsearch o Logstash para identificar documentos.

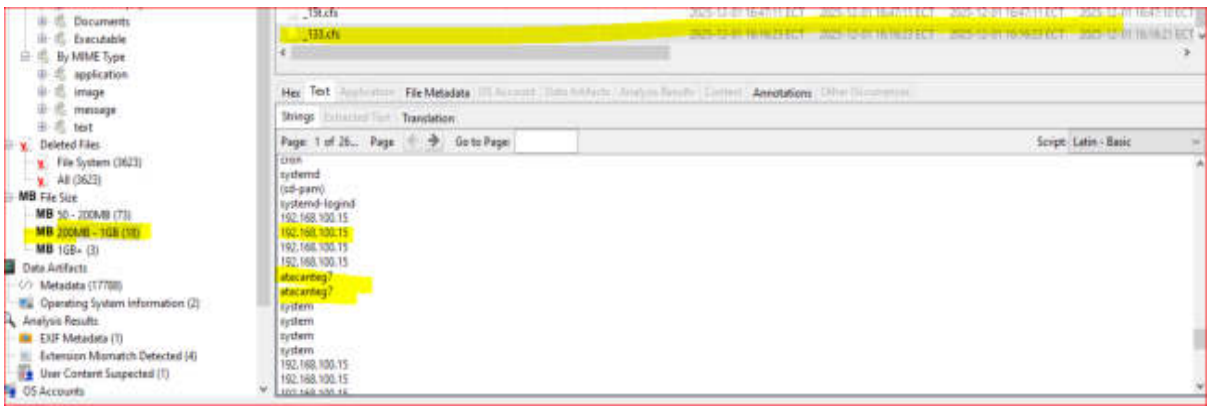
No contienen información legible directa, pero sirven para trazar eventos en el índice.

Contiene la estructura de Lucene, metadatos de términos, listas de postings y referencias a documentos, la información legible que sí aparece es metadatos de eventos como usuarios, IPs, procesos, tipos de log y resultados de autenticación.

No contiene datos sensibles directamente legibles, pero sí nombres de usuarios y eventos que podrían relacionarse con la actividad de un posible atacante.

Figura 160

Verificación de posibles datos sensibles en imagen



Al analizar el contenido del archivo **unalloc_507208_44182695936_44736446464**, podemos ver varios detalles importantes respecto a usuarios y contraseñas, nombre de host, versión, kernel, plataforma utilizada:

Usuarios detectados

En los logs se identifica un usuario explícito posiblemente un usuario sospechoso:

```
New 105esión 7 of user atacanteg7
```

Esto indica que hay un usuario llamado **atacanteg7** registrado en el sistema, se está iniciando una sesión de login para este usuario (via systemd-logind).

Contraseñas

En este archivo aún no hay evidencia de contraseñas, los logs del sistema (auth.log, syslog) no muestran contraseñas por defecto

Solo registran eventos de autenticación, inicios de sesión, IDs de sesión, usuarios y terminales.

A menos que alguien haya configurado un logging inseguro que es muy raro y peligroso

Otros datos de usuario / sesión

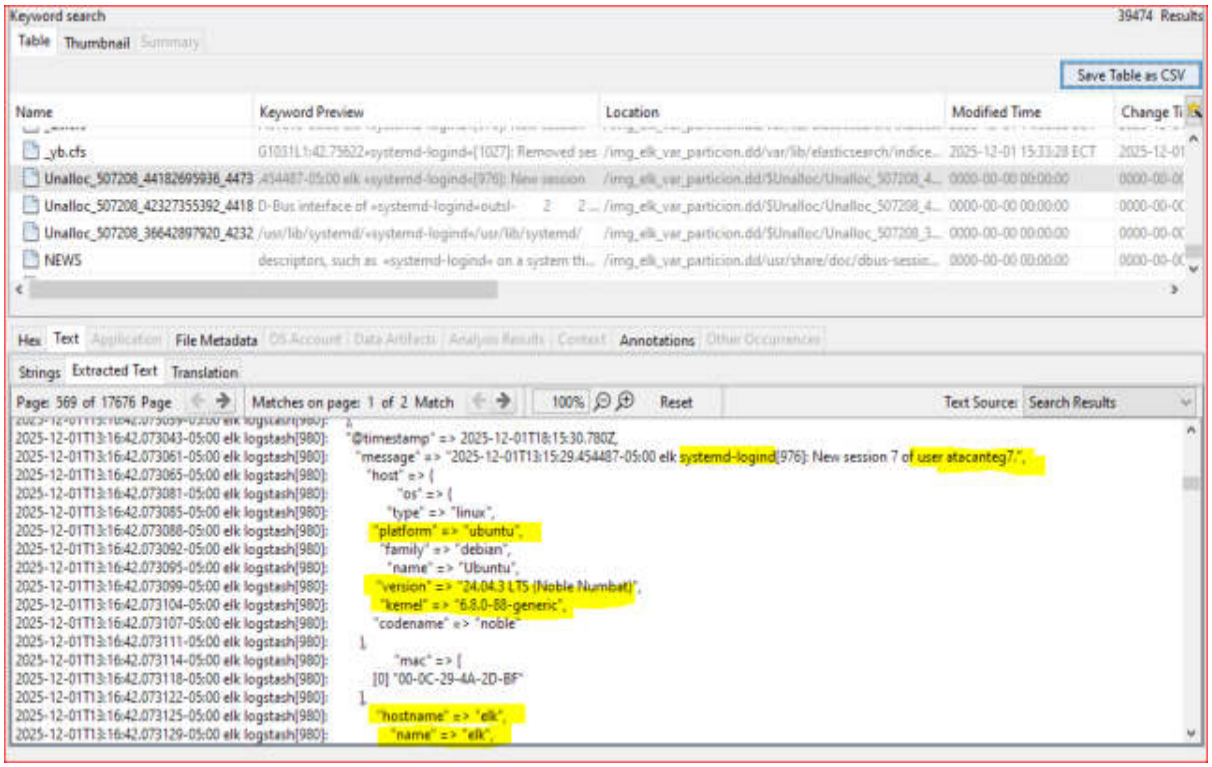
Se registran datos adicionales del host, kernel, arquitectura:

- ✓ Platform: ubuntu

- ✓ Family: debian
- ✓ Name: Ubuntu
- ✓ version: 24.04.3 LTS (Noble Numbat)
- ✓ kernel: 6.8.0-88-generic
- ✓ codename: noble
- ✓ mac: 00-0C-29-4A-2D-BF
- ✓ hostname: elk
- ✓ name: elk
- ✓ architecture: x86_64

Figura 161

Datos adicionales del host en Autopsy



Analizando el archivo `unalloc_507208_44744835072_45687554048`, se puede confirmar que existen inicios de sesión y contraseñas:

Creación de usuario y contraseña:

Se creó un usuario llamado elk con UID 1000 y agregado a varios grupos (adm, sudo, dip, plugdev, lxd).

La contraseña del usuario fue cambiada por root inmediatamente, esto se registra en la fecha

```
2025-11-28T23:22:16.174039+00:00 elk passwd[1255]: password for 'elk' changed by 'root'
```

No se muestra la contraseña real en los logs, lo cual es normal en sistemas Linux: el archivo de logs nunca contiene la contraseña en texto plano.

Inicios de sesión locales y SSH:

El usuario elk abrió sesión localmente y a través de SSH el:

2025-11-28T23:22:26.589319+00:00 elk login[1433]: pam_unix(login:session): session opened for user elk(uid=1000) by elk(uid=0)

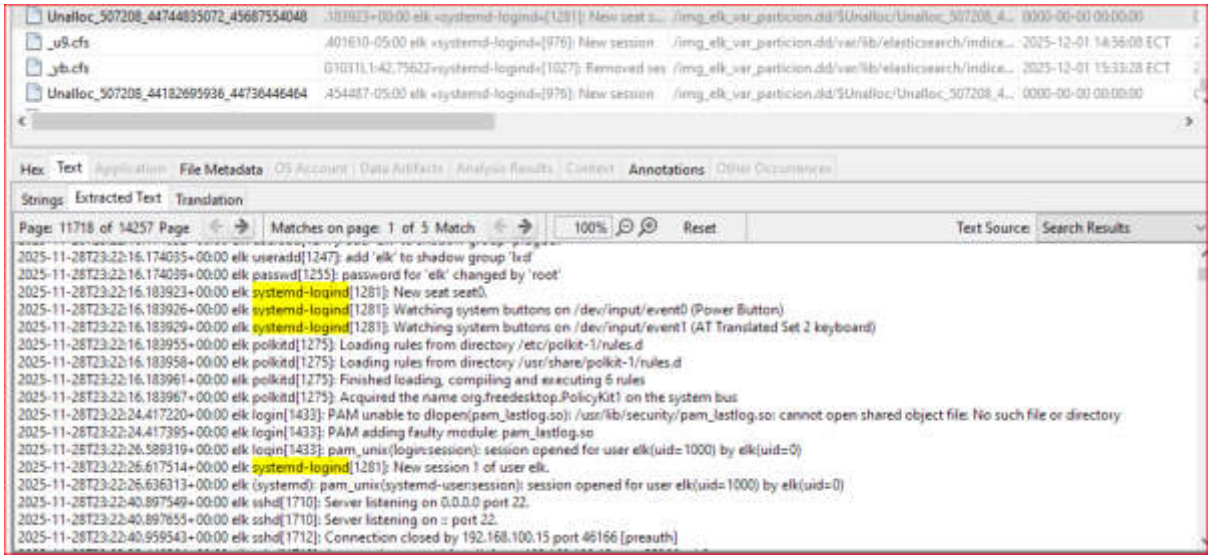
2025-11-28T23:23:32.446564+00:00 elk sshd[1719]: Accepted password for elk from 192.168.100.15 port 55366 ssh2

Esto confirma inicios de sesión exitosos, tanto locales como remotos, pero sin exponer contraseñas, el usuario elk ejecutó múltiples comandos como root usando sudo, por ejemplo: apt update, apt install curl, ufw allow 22/tcp, etc.

Los logs muestran qué comando se ejecutó, quién lo ejecutó, y la sesión de sudo, pero nunca la contraseña de sudo en texto plano.

Figura 162

Vista de logs en Autopsy



Abrimos el archivo **unalloc_507208_32619134976_33696092160** y pudimos detectar que se tiene registros de payloads fallidos de la ip 192.168.100.74 (ubuntu)

Este archivo corresponde a logs de:

Logstash: Procesamiento y recepción de logs y métricas.

Kibana: Visualización y gestión de índices/plugins.

Auditbeat / Syslog / Beats: Ingesta de datos de los agentes hacia Logstash.

Los logs muestran operación normal en muchos casos, pero también hay alertas y errores recurrentes.

Análisis por componente

Logstash

Observaciones

Conexión con Elasticsearch

Restored connection to ES instance {:url=>"http://elastic:xxxxxx@localhost:9200/"}

Logstash logra conectarse y enviar datos a Elasticsearch.

Errores de permisos en syslog

```
syslog listener died {:protocol=>:tcp, :address=>"0.0.0.0:514",
:exception=>#<Errno::EACCES: Permission denied ...>}
```

```
syslog listener died {:protocol=>:udp, :address=>"0.0.0.0:514",
:exception=>#<Errno::EACCES: Permission denied ...>}
```

Existen Problemas recurrentes:

Logstash no puede abrir el puerto 1024 sin permisos de root, esto afecta la ingestión de logs vía syslog TCP/UDP en el puerto 514.

Logstash está recibiendo correctamente datos desde Filebeat/Metricbeat.

Métricas internas (RetentionWindow / JVM collectors)

RetentionWindow{policy=current id=...} forced-compaction result

collector name {:name=>"G1 Young Generation"}

Información de debug sobre JVM y métricas internas.

Normal y no crítico.

Cgroup warnings

One or more required cgroup files or directories not found: /proc/self/cgroup, /sys/fs/cgroup/cpuacct, /sys/fs/cgroup/cpu

Logstash no encuentra ciertos cgroups (usualmente en contenedores o VMs antiguas).

No se detiene la operación, pero limita las métricas de recursos.

Kibana

Observaciones

APM plugin timeout

TimeoutError: Request timed out
Could not create index template: 'apm-source-map'. Retrying...

Problema recurrente:

Kibana intenta crear la plantilla de índice APM y falla por timeout.

Posible causa: Elasticsearch muy ocupado o con permisos insuficientes para crear índices.

Finalmente Kibana se inicia correctamente y está operativo.

Beats / Syslog Inputs

Los logs muestran que payloads de Beats están llegando correctamente:

IP local: 192.168.100.71:5044, IP remote: 192.168.100.74:38572] Received a new payload

Syslog TCP/UDP falla por permisos en el puerto 514.

Tabla 1

Problemas críticos identificados

NAME,"KEYWORD PREVIEW",LOCATION,"MODIFIED TIME","CHANGE TIME","ACCESS TIME","CREATED TIME",SIZE,"FLAGS(DIR)","FLAGS(META)",KNOWN,"MD5 HASH","SHA-256 HASH","MIME TYPE",EXTENSION	
KIBANA_FRAMEWORK_ADAPTER.JS,"""${CONFIG.METHOD}"" IS NOT AN Â«ACCEPTEDÂ« METHOD`)	} } async","/img_elk_var_particion.dd/usr/share/kibana/node_modules/@kbn/metrics-data-access-plugin/server/lib/adapters/framework/kibana_framework_adapter.js","2025-11-07 08:38:35 ECT","2025-11-28 18:55:03 ECT","2025-12-01 07:41:40 ECT","2025-11-28 18:54:50 ECT",5289,Allocated,Allocated,unknown,"f4e9a782ffd063df8b99e550f260c28a","efeb4b14eb63657a54fe0f262f2d586cd3ac416a3d61f38d50ded77dd9b847ad",text/javascript,js
SG_DD.8,"NOTRUNC	

THIS CONVERSION IS Â«ACCEPTEDÂ« FOR COMPATIBILITY WITH","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/MAN/MAN8/SG_DD.8.GZ/SG_DD.8","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00",27874,ALLOCATED,ALLOCATED,UNKNOWN,"CE49F408FA0207A08B5819C2DA50BAFA","84411B5BD1A11F4184C391B9665820885A383B3CBC18959E0F0F7224FCD65EFB",TEXT/PLAIN,8	
TEST_FILEDESCRIPTOR.PY,"C{BYTES} WHICH HAVE BEEN Â«ACCEPTEDÂ« AS WRITTEN.	
@IVAR","/IMG_ELK_VAR_PARTICION.DD/USR/LIB/PYTHON3/DIST-PACKAGES/TWISTED/INTERNET/TEST/TEST_FILEDESCRIPTOR.PY","2025-08-05 12:14:03 ECT","2025-11-28 18:17:40 ECT","2025-11-28 18:17:40 ECT","2025-11-28 18:17:40 ECT",2761,ALLOCATED,ALLOCATED,UNKNOWN,"50B88BD86E5238B778978CF6A00344F5","404FB944637BAAA36817B12C6CD6689868F9B3C7474DE82695654FDC3D822C24",TEXT/X-PYTHON,PY	
SERDEV.H,"RETURNS NUMBER OF BYTES Â«ACCEPTEDÂ«	may sleep.
*** @WRITE_WAKEUP:","/IMG_ELK_VAR_PARTICION.DD/USR/SRC/LINUX-HEADERS-6.8.0-88/INCLUDE/LINUX/SERDEV.H","2024-03-10 15:38:09 ECT","2025-11-28 18:20:17 ECT","2025-11-28 18:20:05 ECT","2025-11-28 18:20:15 ECT",10363,ALLOCATED,ALLOCATED,UNKNOWN,"E2707E8EEF71428229ACC9F78B396CAF","3DD316AB3B16E2EF6BDCA3DC82C93A27E1739E5BFD75ED4808FB2419333BA349",TEXT/X-CHDR,H**	
LVEXTEND.8,"SPECIFIC TYPE, WHERE THE	
Â«ACCEPTEDÂ« LV TYPES ARE LISTED.","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/MAN/MAN8/LVEXTEND.8.GZ/LVEXTEND.8","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00",19010,ALLOCATED,ALLOCATED,UNKNOWN,"6F5CA57C3EE7D14B3FDBF83501400A9E","72213A2014AB8BB1AEFC6BD1DBCD69829B69E252E36C0EFF040E8A2171831B18",TEXT/PLAIN,8	

JAVA.1,"BOTH THE SOURCE VERSION Â«ACCEPTEDÂ« BY COMPILER AND THE SYSTEM","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/ELASTICSEARCH/JDK/MAN/MAN1/JAVA.1","2025-11-07 08:41:36 ECT","2025-11-28 18:26:05 ECT","2025-11-28 18:25:55 ECT","2025-11-28 18:26:05 ECT",201578,ALLOCATED,ALLOCATED,UNKNOWN,"AFD6338DB2995445F436C21F1AE43FFB","FA91B8283AA0AA1B749804B5C12FAC2FF4ACE83A74E59FC1D6BA85913484C24A",TEXT/TROFF,1

CRYPTSETUP-LUKSADDKEY.8,"LUKS1, ONLY PBKDF2 IS Â«ACCEPTEDÂ« (NO NEED TO USE THIS","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/MAN/MAN8/CRYPTSETUP-LUKSADDKEY.8.GZ/CRYPTSETUP-LUKSADDKEY.8","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00",15906,ALLOCATED,ALLOCATED,UNKNOWN,"C84C711B19929A2591FEF0F43B2E5AB9","B74E3C4C5B8DF772E9B70A52B558C4F6E4493B4A70D390757460143D102E9449",TEXT/TROFF,8

JDB.1,"FOLLOWING OPTIONS ARE Â«ACCEPTEDÂ« BY THE \F[V]JDB\F[R]","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/ELASTICSEARCH/JDK/MAN/MAN1/JDB.1","2025-11-07 08:41:36 ECT","2025-11-28 18:26:05 ECT","2025-11-28 18:25:55 ECT","2025-11-28 18:26:05 ECT",9389,ALLOCATED,ALLOCATED,UNKNOWN,"495539A350348870666907632FF0F85B","71B2F29C433BA253095007A0170703FB2C6EA236B7F884219369BDB9D21F77A6",TEXT/TROFF,1

FIELDS.YML,"EXAMPLE: Â«ACCEPTEDÂ«

DESCRIPTION:","/IMG_ELK_VAR_PARTICION.DD/ETC/FILEBEAT/FIELDS.YML","2025-11-07 07:24:10 ECT","2025-11-28 22:12:25 ECT","2025-11-28 22:12:25 ECT","2025-11-28 22:12:24 ECT",1219302,ALLOCATED,ALLOCATED,UNKNOWN,"33D7ECB2369B33B1DE8A25ADD963BEC8","CD384D38DA3EB6E853D2A10A7FDE2C182AAD43F396EB3FC022DD2FB9AD86598E",TEXT/X-YAML,YML

XMLPARSER.JS,"NEW ERROR(""XML DATA IS Â«ACCEPTEDÂ« IN STRING OR BYTES[]","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/KIBANA/NODE_MODULES/FAST-XML-PARSER/SRC/V5/XMLPARSER.JS","2025-11-07 08:37:48 ECT","2025-11-28 18:55:04 ECT","2025-11-28 18:54:38 ECT","2025-11-28 18:55:02 ECT",3103,ALLOCATED,ALLOCATED,UNKNOWN,"9D6BA8027068B25DBCD37B8F32AAF4D5","CEDE2B29DA212BD8985F20E73073484840DFDE19F1FFE82814BE29E1885C2F3A",TEXT/JAVASCRIPT,JS	
FILEBEAT.REFERENCE.YML,"ELASTICSEARCH OUTPUT ARE Â«ACCEPTEDÂ« HERE AS WELL. #	
NOTE","/IMG_ELK_VAR_PARTICION.DD/ETC/FILEBEAT/FILEBEAT.REFERENCE.YML","2025-11-07 07:24:10 ECT","2025-11-28 22:12:25 ECT","2025-11-28 22:12:25 ECT","2025-11-28 22:12:24 ECT",176274,ALLOCATED,ALLOCATED,UNKNOWN,"9C13E6D4373008D8347F36A787D85FFE","B92A1EFE94A0254FDCA412446EB8B0A4BCB4B0F61014ECFD2E240236EE671696",TEXT/X-YAML,YML	
SG_WRITE_X.8,"ARE ZEROS. THE FORMAT Â«ACCEPTEDÂ«	
IS RELATIVELY LOOSE WITH","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/MAN/MAN8/SG_WRITE_X.8.GZ/SG_WRITE_X.8","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00",32983,ALLOCATED,ALLOCATED,UNKNOWN,"B42D76158A5CB27634C2434BACBFFAEE","BA833231813C64B3917E1EA8505DC44B9DADE12245D49017E27CA7F2F1D20A89",TEXT/PLAIN,8	
CMP.H,"::= INTEGER {	

* Â«ACCEPTEDÂ« (0),","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/KIBANA/NODE/GLIBC-217/INCLUDE/NODE/OPENSSL/ARCHS/VC-WIN64-ARM/NO-ASM/INCLUDE/OPENSSL/CMP.H","2025-11-07 08:39:12 ECT","2025-11-28 18:55:03 ECT","2025-11-28 18:54:38 ECT","2025-11-28 18:54:39 ECT",41123,ALLOCATED,ALLOCATED,UNKNOWN,"0004A88D9AC2F2897FE756B3780DF38F","28ABBCA1B6798A226A323C5B23DD884D3FE739F5D9C8B0F1329CC665355D6317",TEXT/X-CHDR,H	
GOVERNANCE.MD,"MUST BE REVIEWED AND Â«ACCEPTEDÂ« BY A COLLABORATOR WITH","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/KIBANA/NODE_MODULES/STREAM-CHOPPER/NODE_MODULES/READABLE-STREAM/GOVERNANCE.MD","2025-11-07 08:37:48 ECT","2025-11-28 18:55:04 ECT","2025-11-28 18:54:38 ECT","2025-11-28 18:54:58 ECT",5550,ALLOCATED,ALLOCATED,UNKNOWN,"70B44945CEC4643CA805D87F673FBD34","7A521E462D1C6F3B599C44637FB337BBF969DDA311510A87236EC539A415331D",TEXT/X-WEB-MARKDOWN,MD	
PERMESSAGE-DEFLATE.JS,"E * @RETURN {OBJECT} Â«ACCEPTEDÂ« CONFIGURATION * @PUBLIC","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/KIBANA/NODE_MODULES/WS/LIB/PERMESSAGE-DEFLATE.JS","2025-11-07 08:37:39 ECT","2025-11-28 18:55:04 ECT","2025-12-01 07:41:15 ECT","2025-11-28 18:55:00 ECT",14510,ALLOCATED,ALLOCATED,UNKNOWN,"15345EF613B64AE83C6BA7901F535397","ADCEDCB3069B7C6DB3FD6CEC93699EF70BD4061126B987DD9A8F7CB16D2B7530",TEXT/JAVASCRIPT,JS	

MOUNT.8,"\FBUSER\FP MOUNT OPTION IS Â«ACCEPTEDÂ« IF NO USERNAME IS SPECIFIED","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/MAN/MAN8/MOUNT.8.GZ/MOUNT.8","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00",108503,ALLOCATED,ALLOCATED,UNKNOWN,"7BB9745D27086E3B030BF2239BA2637D","F86E91FA582ECEAFDFDB8527C1DC880EA272195E6401B743F72FA5C84C89329A",TEXT/TROFF,8	
CLIENT.RB,"LIST OF THE	
# Â«ACCEPTEDÂ« MODES AND THE CONFIGURATION","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/LOGSTASH/VENDOR/BUNDLE/JRUBY/3.1.0/GEMS/AWS-SDK-CORE-3.226.3/LIB/AWS-SDK-SSOOIDC/CLIENT.RB","2025-11-04 13:09:44 ECT","2025-11-28 21:59:32 ECT","2025-11-28 21:59:24 ECT","2025-11-28 21:59:25 ECT",55888,ALLOCATED,ALLOCATED,UNKNOWN,"E3CFAC48290158F73BBF54DAACB3AF46","585613999D77A6C4FE69C9F1C2191012EB9138537ED89EDE89D680EB73C74C8A",TEXT/X-RUBY,RB	
TRANSFORM.CHUNK.409.JS,"FOR IPV4 ADDRESSES THE Â«ACCEPTEDÂ« RANGE IS [0, 32].\\NFOR","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/KIBANA/NODE_MODULES/@KBN/TRANSFORM-PLUGIN/TARGET/PUBLIC/TRANSFORM.CHUNK.409.JS","2025-11-07 08:38:03 ECT","2025-11-28 18:55:03 ECT","2025-11-28 18:54:38 ECT","2025-11-28 18:54:53 ECT",444846,ALLOCATED,ALLOCATED,UNKNOWN,"CE0C5E6D54F6AFCEABBF31874512265A","BE62C95F5F657492829BD8F197176F3BBFA5A5BDAF95CC15746C298301BE0168",TEXT/JAVASCRIPT,JS	
JMOD.1,"CREATING THE JMOD FILE.	

THE Â«ACCEPTEDÂ« VALUES ARE \F[V]ZIP-","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/ELASTICSEARCH/JDK/MAN/MAN1/JMOD.1","2025-11-07 08:41:36 ECT","2025-11-28 18:26:05 ECT","2025-11-28 18:25:55 ECT","2025-11-28 18:26:05 ECT",11303,ALLOCATED,ALLOCATED,UNKNOWN,"1631E89CC4E5B5EF246A90DA52762734","4348B7E02F059E37AA4034B8436CADF36CDE83B73785B8C81BA1B9BC616F661A",TEXT/TROFF,1	
SAMPLE.HTTP.YML.DISABLED,"NOT	
# BEING 404 IS Â«ACCEPTEDÂ«.	
#STATUS: 0","/IMG_ELK_VAR_PARTICION.DD/HOME/ELK/ELASTIC-AGENT-8.11.1-LINUX-X86_64/DATA/ELASTIC-AGENT-03EF9D/COMPONENTS/MONITORS.D/SAMPLE.HTTP.YML.DISABLED","2023-11-11 08:27:03 ECT","2025-12-01 08:54:04 ECT","2025-12-01 09:05:31 ECT","2025-12-01 08:54:04 ECT",2850,ALLOCATED,ALLOCATED,UNKNOWN,"DED6FAFCEF3AE725809A30D76E5D5A6C","7746F621F248875468D8DD9FED6E0DCCB323B88E2F382A4EC854354133186D92",TEXT/PLAIN,DISABLED	
WEBSOCKET-SERVER.JS,"{OBJECT} EXTENSIONS THE Â«ACCEPTEDÂ« EXTENSIONS * @PARAM {STRING}","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/KIBANA/NODE_MODULES/WS/LIB/WEBSOCKET-SERVER.JS","2025-11-07 08:37:39 ECT","2025-11-28 18:55:04 ECT","2025-12-01 07:41:15 ECT","2025-11-28 18:55:00 ECT",16393,ALLOCATED,ALLOCATED,UNKNOWN,"DCE6F00DA0B660CA42CC383CD9E876EA","FE8A410E89325608DC94D09778A2650BF041B9B7F5DFF61D5005EA111E09A508",TEXT/JAVASCRIPT,JS	

_STREAM_WRITABLE.JS,"THAT `NULL` IS NEVER Â«ACCEPTEDÂ« // AND UNDEFINED/NON-STRING","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/KIBANA/NODE_MODULES/STREAM-CHOPPER/NODE_MODULES/READABLE-STREAM/LIB/_STREAM_WRITABLE.JS","2025-11-07 08:38:03 ECT","2025-11-28 18:55:04 ECT","2025-12-01 07:39:29 ECT","2025-11-28 18:54:58 ECT",21907,ALLOCATED,ALLOCATED,UNKNOWN,"1997E9040E13749540039901789AFAC1","0F3F2ACCECD343DFAE91D7DAA72F3AD125B6F139D5883FF51362042D8CB65CF6",TEXT/JAVASCRIPT,JS	
SAMPLE.TCP.YML.DISABLED,"RESPONSE WILL BE # Â«ACCEPTEDÂ« AS OK. IF ONLY `RECE","/IMG_ELK_VAR_PARTICION.DD/HOME/ELK/ELASTIC-AGENT-8.11.1-LINUX-X86_64/DATA/ELASTIC-AGENT-03EF9D/COMPONENTS/MONITORS.D/SAMPLE.TCP.YML.DISABLED","2023-11-11 08:27:04 ECT","2025-12-01 08:54:04 ECT","2025-12-01 09:05:31 ECT","2025-12-01 08:54:04 ECT",3526,ALLOCATED,ALLOCATED,UNKNOWN,"838FB2EDE909F662CBBF422BF6A61A00","486B3960A308309DA9EE8A9C536496179848442145F63C71E33C0BCCF93D729A",TEXT/PLAIN,DISABLED	
JSHELL.1,"AND EXPRESSIONS ARE Â«ACCEPTEDÂ«. THE BITS OF CODE ENTERED","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/ELASTICSEARCH/JDK/MAN/MAN1/JSHELL.1","2025-11-07 08:41:36 ECT","2025-11-28 18:26:05 ECT","2025-11-28 18:25:55 ECT","2025-11-28 18:26:05 ECT",41952,ALLOCATED,ALLOCATED,UNKNOWN,"B02BFFB070EA59DEFA5A95A1BACD4CB9","A9D26A4B2745AECCCB1EDF2664ADCADE927E5AF86B30139D97DF974FFC775E14",TEXT/TROFF,1	
CLIENT.RB,"A LIST OF THE Â€ ÂŒ Â€ Â€	

Â«ACCEPTEDÂ« MODES AND THE CONFIGURATION","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/LOGSTASH/VENDOR/BUNDLE/JRUBY/3.1.0/GEMS/AWS-SDK-CLOUDFRONT-1.119.0/LIB/AWS-SDK-CLOUDFRONT/CLIENT.RB","2025-11-04 13:09:45 ECT","2025-11-28 21:59:32 ECT","2025-11-28 21:59:24 ECT","2025-11-28 21:59:25 ECT",722825,ALLOCATED,ALLOCATED,UNKNOWN,"F7C4DA7FA0610580E905EF1B4E58B021","973A0EB6508FAD2E900627765A2A297101F76711E50CAFB03DBC347575DF765F",TEXT/X-RUBY,RB	
SERVICE-2.JSON,"IES</CODE>. THE ONLY Â«ACCEPTEDÂ« VALUE FOR THIS PARAMETER","/IMG_ELK_VAR_PARTICION.DD/USR/LIB/PYTHON3/DIST-PACKAGES/BOTOCORE/DATA/TRANSCRIBE/2017-10-26/SERVICE-2.JSON","2025-08-05 12:14:02 ECT","2025-11-28 18:17:37 ECT","2025-11-28 18:17:37 ECT","2025-11-28 18:17:37 ECT",272683,ALLOCATED,ALLOCATED,UNKNOWN,"53B11D4A65467924F7920D4E2C264067","13E10B817623C41995A0408DAFF1F15BAD333AA418E0200E6048825A12CD2507",TEXT/PLAIN,JSON	
OSQUERYBEAT,"NILCONFIG DOES NOT MATCH Â«ACCEPTEDÂ« CONFIGURATIONSCONFIGURATION","/IMG_ELK_VAR_PARTICION.DD/HOME/ELK/ELASTIC-AGENT-8.11.1-LINUX-X86_64/DATA/ELASTIC-AGENT-03EF9D/COMPONENTS/OSQUERYBEAT","2023-11-11 08:27:09 ECT","2025-12-01 08:54:08 ECT","2025-12-01 09:05:31 ECT","2025-12-01 08:54:04 ECT",108099512,ALLOCATED,ALLOCATED,UNKNOWN,"FCCDC694CF14BBCD501AF0E90730CFC1","4D0BAAEED6BB6B1D4BD738663AFF7D32BDAD6335FD8A200EF30DDBAC8F298DE5",APPLICATION/X-SHAREDLIB,	
OUTPUT.JAVA,"ENCODING - ANY VALUE Â«ACCEPTEDÂ« HERE	

_ENCODING","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/LOGSTASH/JDK/LIB/SRC.ZIP/JAVA.XML/COM/SUN/ORG/APACHE/XALAN/INTERNAL/XSLTC/COMPILER/OUTPUT.JAVA","2025-10-21 00:00:00 ECT","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00",15735,ALLOCATED,ALLOCATED,UNKNOWN,"2CDF1EF3A8AE742FB2E4CB46861EBB1B","184544E8DC5E772F99B71B7AD5F87D516ECCFFC625E486D1AD8F0ED1E47E0350",TEXT/X-JAVA-SOURCE,JAVA

OSQUERYBEAT.REFERENCE.YML,"ELASTICSEARCH OUTPUT ARE Â«ACCEPTEDÂ« HERE AS WELL.
#
NOTE","/IMG_ELK_VAR_PARTICION.DD/HOME/ELK/ELASTIC-AGENT-8.11.1-LINUX-X86_64/DATA/ELASTIC-AGENT-03EF9D/COMPONENTS/OSQUERYBEAT.REFERENCE.YML","2023-11-11 08:27:09 ECT","2025-12-01 08:54:08 ECT","2025-12-01 09:05:41 ECT","2025-12-01 08:54:08 ECT",46478,ALLOCATED,ALLOCATED,UNKNOWN,"6E81207121E622E77933542CD8C323B8","5CB23D2B760C3798CF0449BE80D80557FB1BA139D80E0C999A2BC3D4B9AFD1D6",TEXT/X-YAML,YML

OSQUERYBEAT.YML,"ELASTICSEARCH OUTPUTS ARE Â«ACCEPTEDÂ« HERE AS WELL.
#
NOTE","/IMG_ELK_VAR_PARTICION.DD/HOME/ELK/ELASTIC-AGENT-8.11.1-LINUX-X86_64/DATA/ELASTIC-AGENT-03EF9D/COMPONENTS/OSQUERYBEAT.YML","2023-11-11 08:27:09 ECT","2025-12-01 08:54:08 ECT","2025-12-01 09:05:41 ECT","2025-12-01 08:54:08 ECT",6505,ALLOCATED,ALLOCATED,UNKNOWN,"5934A5A6FEC01C8A895C915CD60FFB93","F26630D003DE870F7AAD288D27D2C709894E2B15C33DF286F70DC4910D33C3FD",TEXT/X-YAML,YML
OSQUERYD,"BE MONITORING THE LV.
Â«ACCEPTEDÂ« VALUES:

REMOVE","/IMG_ELK_VAR_PARTICION.DD/HOME/ELK/ELASTIC-AGENT-8.11.1-LINUX-X86_64/DATA/ELASTIC-AGENT-03EF9D/COMPONENTS/OSQUERYD","2023-11-11 08:27:10 ECT","2025-12-01 08:54:16 ECT","2025-12-01 09:05:41 ECT","2025-12-01 08:54:08 ECT",268531864,ALLOCATED,ALLOCATED,UNKNOWN,"73A7B75FFF981E31CBA302F6BCDEB5C3","BB18C4E00D8BE5A11764FDFD4D42AC8B51B3E022A91CFBCD533FA7546A6E9F03",APPLICATION/X-SHAREDLIB,
DPKG.STATUS.0,"IS
ONE OF THE FORMATS Â«ACCEPTEDÂ« FOR INLINE IMAGES BY","/IMG_ELK_VAR_PARTICION.DD/VAR/BACKUPS/DPKG.STATUS.0","2025-11-30 17:43:16 ECT","2025-12-01 07:38:26 ECT","2025-11-30 17:43:16 ECT","2025-12-01 07:38:25 ECT",732105,ALLOCATED,ALLOCATED,UNKNOWN,"8F1BB865C840FD4C87BE028877B00523","8FBBFAEA50A0E707C44916778581C12903A40BA1528390DC2F42CD4B6BE13FE5",TEXT/PLAIN,0
TEST_TCP.PY,"HAS BEEN
Â«ACCEPTEDÂ«.

""""""""""","/IMG_ELK_VAR_PARTICION.DD/USR/LIB/PYTHON3/DIST-PACKAGES/TWISTED/INTERNET/TEST/TEST_TCP.PY","2025-08-05 12:14:03 ECT","2025-11-28 18:17:40 ECT","2025-11-28 18:17:40 ECT","2025-11-28 18:17:40 ECT",108237,ALLOCATED,ALLOCATED,UNKNOWN,"171F5D12E96BF579807705917C49716E","41B17EEB0FDBEA6628D88D00AF8733D5208E0B006BB86BAF3912BF6AF1B0F886",TEXT/X-PYTHON,PY

PACKETBEAT,"NILCONFIG DOES NOT MATCH Â«ACCEPTEDÂ« CONFIGURATIONSCONFIGURATION","/IMG_ELK_VAR_PARTICION.DD/HOME/ELK/ELASTIC-AGENT-8.11.1-LINUX-X86_64/DATA/ELASTIC-AGENT-03EF9D/COMPONENTS/PACKETBEAT","2023-11-11 08:27:11 ECT","2025-12-01 08:54:19 ECT","2025-12-01 09:06:12 ECT","2025-12-01 08:54:16 ECT",113780792,ALLOCATED,ALLOCATED,UNKNOWN,"CA476F9CC6CFB8D0BB0195A9CF3BBED4","F49C368170F7753258FEC8328C21629EA6EA068B7E8453E0590349EF5829C16B",APPLICATION/X-SHAREDLIB,	
PACKETBEAT.REFERENCE.YML,"TYPE. CURRENTLY THE ONLY Â«ACCEPTEDÂ« VALUE IS BINARY FOR","/IMG_ELK_VAR_PARTICION.DD/HOME/ELK/ELASTIC-AGENT-8.11.1-LINUX-X86_64/DATA/ELASTIC-AGENT-03EF9D/COMPONENTS/PACKETBEAT.REFERENCE.YML","2023-11-11 08:27:11 ECT","2025-12-01 08:54:19 ECT","2025-12-01 09:06:24 ECT","2025-12-01 08:54:19 ECT",86033,ALLOCATED,ALLOCATED,UNKNOWN,"1667DB03FBFDB2E129C0021A4C1BCDEE","034C75F259E06AE371C2ECB0AAF3460DC1CE08F2B96A37DD2C595FFBDDDF3E90",TEXT/X-YAML,YML	
RAWHTML.JAVA,"* THE STRING IS Â«ACCEPTEDÂ« AS-IS AND IS NOT VALIDATED","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/LOGSTASH/JDK/LIB/SRC.ZIP/JDK.JAVADOC/JDK/JAVADOC/INTERNAL/DOCLETS/FORMATS/HTML/MARKUP/RAWHTML.JAVA","2025-10-21 00:00:00 ECT","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00",6099,ALLOCATED,ALLOCATED,UNKNOWN,"2F53467B773519780AFA86A1A429D663","EF3D32B2E71A0F234353C5991F54A110F1F5946060C18F7E425F30976F6BB0F9",TEXT/X-JAVA-SOURCE,JAVA	

GET_TLS_OPTIONS.JS,"`TLSOPTIONS` THAT ARE Â«ACCEPTEDÂ« BY THE HAPI SERVER, *","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/KIBANA/NODE_MODULES/@KBN/SERVER-HTTP-TOOLS/SRC/GET_TLS_OPTIONS.JS","2025-11-07 08:37:47 ECT","2025-11-28 18:55:03 ECT","2025-12-01 07:39:35 ECT","2025-11-28 18:54:51 ECT",1393,ALLOCATED,ALLOCATED,UNKNOWN,"971D5A56A7B9E1B8ED709C7B28F6D130","F71D0AA2D84AF6C866C472D24F10933A740A6DB7CC5DD3B3529307BD96F21C05",TEXT/JAVASCRIPT,JS
PACKETBEAT.YML,"ELASTICSEARCH OUTPUTS ARE Â«ACCEPTEDÂ« HERE AS WELL. #
NOTE","/IMG_ELK_VAR_PARTICION.DD/HOME/ELK/ELASTIC-AGENT-8.11.1-LINUX-X86_64/DATA/ELASTIC-AGENT-03EF9D/COMPONENTS/PACKETBEAT.YML","2023-11-11 08:27:11 ECT","2025-12-01 08:54:19 ECT","2025-12-01 09:06:24 ECT","2025-12-01 08:54:19 ECT",11487,ALLOCATED,ALLOCATED,UNKNOWN,"160CDC5D42B9B027ED2D37FDAEA9CD0B","B3B8FED78C1FC8DA436277CA5C6B5C12E9C8676FA50B1AE265507695C0B30DF6",TEXT/X-YAML,YML
GET_SERVER_OPTIONS.JS,"`SERVEROPTIONS` THAT ARE Â«ACCEPTEDÂ« BY THE HAPI SERVER. */","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/KIBANA/NODE_MODULES/@KBN/SERVER-HTTP-TOOLS/SRC/GET_SERVER_OPTIONS.JS","2025-11-07 08:37:47 ECT","2025-11-28 18:55:03 ECT","2025-12-01 07:39:35 ECT","2025-11-28 18:54:51 ECT",2171,ALLOCATED,ALLOCATED,UNKNOWN,"C7CB3B5CFF0F2F2BB26F9C5360CD28A6","8534F6EC606EA9EFE3B22FD288BA976DFCD2261EFC8979B227316F2CA1C21D4F",TEXT/JAVASCRIPT,JS
STREAMPRINTSERVICEFACTORY.JAVA,"DOCUMENT FLAVORS THAT CAN BE Â«ACCEPTEDÂ« BY

*** PRINTERS OBTAINED","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/LOGSTASH/JDK/LIB/SRC.ZIP/JAVA.DESKTOP/JAVAX/PRINT/STREAMPRINTSERVICEFACTORY.JAVA","2025-10-21 00:00:00 ECT","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00",11140,ALLOCATED,ALLOCATED,UNKNOWN,"42C9A0BC2321DC232978494735A7015D","6E6D3E1759C25D5CAF6149E1C26EA8EADCC79BA5C398FC6513901A102F78DD4E",TEXT/X-JAVA-SOURCE,JAVA**	
LIBAPT-PKG6.0.MO,"'%S' MEMBER	
THIS MUST BE Â«ACCEPTEDÂ« EXPLICITLY BEFORE UPDATES","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/LOCALE/ZH_CN/LC_MESSAGES/LIBAPT-PKG6.0.MO","2025-08-05 12:14:05 ECT","2025-11-28 18:17:57 ECT","2025-11-28 18:17:57 ECT","2025-11-28 18:17:57 ECT",32916,ALLOCATED,ALLOCATED,UNKNOWN,"DDFA709397AFB7E92D463947078AC572","FAE8037426DD0743C13281B5194867DDFF3102104310ADD60524C73FF64A369F",APPLICATION/OCTET-STREAM,MO	
KEYTOOL.1,"01:02:03:04 AND 01020304 ARE Â«ACCEPTEDÂ« AS IDENTICAL VALUES.","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/ELASTICSEARCH/JDK/MAN/MAN1/KEYTOOL.1","2025-11-07 08:41:36 ECT","2025-11-28 18:26:05 ECT","2025-11-28 18:25:55 ECT","2025-11-28 18:26:05 ECT",109595,ALLOCATED,ALLOCATED,UNKNOWN,"851FE5D9D18F678E47E6B9A1B3A3B268","34951E3628616CC8B3BBD3D8F76EA3C50CA6E5E0535ABE7B729522033A76D9DD",TEXT/TROFF,1	

SERVICE-2.JSON,"ACTIVE AND RECYCLED ARE Â«ACCEPTEDÂ« VALUES FROM THE API.</P>""","/IMG_ELK_VAR_PARTICION.DD/USR/LIB/PYTHON3/DIST-PACKAGES/BOTOCORE/DATA/WORKDOCS/2016-05-01/SERVICE-2.JSON","2025-08-05 12:14:02 ECT","2025-11-28 18:17:37 ECT","2025-11-28 18:17:37 ECT","2025-11-28 18:17:37 ECT",147194,ALLOCATED,ALLOCATED,UNKNOWN,"E017C00E35634C3773FD4FCF31F73791","7BAC1E308B6678FDA46FD181847A4F2F3AACAF8D94AE19099B31948DDC87D532",TEXT/PLAIN,JSON

GOVERNANCE.MD,"MUST BE REVIEWED AND Â«ACCEPTEDÂ« BY A COLLABORATOR WITH","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/KIBANA/NODE_MODULES/DUPLEXER2/NODE_MODULES/READABLE-STREAM/GOVERNANCE.MD","2025-11-07 08:37:48 ECT","2025-11-28 18:55:03 ECT","2025-11-28 18:54:38 ECT","2025-11-28 18:54:40 ECT",5550,ALLOCATED,ALLOCATED,UNKNOWN,"70B44945CEC4643CA805D87F673FBD34","7A521E462D1C6F3B599C44637FB337BBF969DDA311510A87236EC539A415331D",TEXT/X-WEB-MARKDOWN,MD

X509_VFY.H,"TRUST IF NO EXPLICIT Â«ACCEPTEDÂ« TRUST EKUS */ # DEFINE","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/KIBANA/NODE/GLIBC-217/INCLUDE/NODE/OPENSSL/ARCHS/VC-WIN64-ARM/NO-ASM/INCLUDE/OPENSSL/X509_VFY.H","2025-11-07 08:39:12 ECT","2025-11-28 18:55:03 ECT","2025-11-28 18:54:38 ECT","2025-11-28 18:54:39 ECT",52027,ALLOCATED,ALLOCATED,UNKNOWN,"03954943980620D1194BEB68249B9EC7","00ECA88DA5CAC83641BA85391C7351252411AF33B9F9BF12A31CDA3D10F5C2F1",TEXT/X-CHDR,H

ELASTIC-AGENT,"INVALID PROTOCOL %S, Â«ACCEPTEDÂ« VALUES ARE 'HTTP' AND","/IMG_ELK_VAR_PARTICION.DD/HOME/ELK/ELASTIC-AGENT-8.11.1-LINUX-X86_64/DATA/ELASTIC-AGENT-03EF9D/ELASTIC-AGENT","2023-11-11 08:30:17 ECT","2025-12-01 08:54:22 ECT","2025-12-01 09:02:40 ECT","2025-12-01 08:54:21 ECT",55374976,ALLOCATED,ALLOCATED,UNKNOWN,"680FDF23290CAE34B3EEF6F0C508787B","166122D5F43FB2BBAD9C8F9ED6870FAEEF5D833DB5E1DDF721A7E240CDAF8945",APPLICATION/X-SHAREDLIB,	
MKE2FS.8,"B \-R	
OPTION IS STILL Â«ACCEPTEDÂ« FOR BACKWARDS COMPATIBILITY","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/MAN/MAN8/MKE2FS.8.GZ/MKE2FS.8","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00",26379,ALLOCATED,ALLOCATED,UNKNOWN,"BCBC28F2354106E22EABF6454BC1A778","A3842E5737EFE06874E2CBEFA97C51385F2C4A6F9D06B63A4765400FB2BBCF79",TEXT/TROFF,8	
_DUMBWIN32PROC.PY,"LINES ARE CULTURALLY	
Â«ACCEPTEDÂ« IN THE WINDOWS WORLD","/IMG_ELK_VAR_PARTICION.DD/USR/LIB/PYTHON3/DIST-PACKAGES/TWISTED/INTERNET/_DUMBWIN32PROC.PY","2025-08-05 12:14:03 ECT","2025-11-28 18:17:40 ECT","2025-11-28 18:17:40 ECT","2025-11-28 18:17:40 ECT",12584,ALLOCATED,ALLOCATED,UNKNOWN,"2515E88EB3CD369964FB4D4BEADCB4D8","3D59D6C2D5AA41D974946D4E9566CAE8A32773100BB3171016AF08A9A26D3C26",TEXT/X-PYTHON,PY	

DPKG-DISTADDFILE.1,"18.5). THE CURRENTLY Â«ACCEPTEDÂ« VALUES ARE: \FBAUTO\FR","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/MAN/IT/MAN1/DPKG-DISTADDFILE.1.GZ/DPKG-DISTADDFILE.1","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00",3380,ALLOCATED,ALLOCATED,UNKNOWN,"F042CAC0946C66501392DB32C1226374","0F58C88C345E88B7A1AB39E6CBB4A7BE224333003F704B62D4563BF1C5869EA2",TEXT/TROFF,1

_STREAM_WRITABLE.JS,"THAT `NULL` IS NEVER Â«ACCEPTEDÂ« // AND UNDEFINED/NON-STRING","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/KIBANA/NODE_MODULES/DUPLEXER2/NODE_MODULES/READABLE-STREAM/LIB/_STREAM_WRITABLE.JS","2025-11-07 08:38:03 ECT","2025-11-28 18:55:03 ECT","2025-12-01 07:40:35 ECT","2025-11-28 18:54:40 ECT",20335,ALLOCATED,ALLOCATED,UNKNOWN,"09B0D94AF81D8A886E8BDDA4E1D72AFE","E6359AC652ED97F5F328C586C7A6B8F163782A9CA13DA476E609A981C75E0469",TEXT/JAVASCRIPT,JS

CDC.H,"SPECIFIES ACM NOTIFICATIONS, Â«ACCEPTEDÂ« BY RNDIS

* RNDIS ALSO","/IMG_ELK_VAR_PARTICION.DD/USR/INCLUDE/LINUX/USB/CDC.H","2025-10-10 18:38:46 ECT","2025-11-28 18:20:07 ECT","2025-11-28 18:20:05 ECT","2025-11-28 18:20:07 ECT",13475,ALLOCATED,ALLOCATED,UNKNOWN,"D0F2C6A5E428707AA23772A561BA95F9","922BD199D3D8163BE1027D3B7A8AB679FD8A7FBA98F1F67F4F597D5E36ADCE81",TEXT/X-CHDR,H

SG_WRITE_VERIFY.8,"AND 3 (INCLUSIVE) ARE Â«ACCEPTEDÂ«. THE DEFAULT IS VALUE","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/MAN/MAN8/SG_WRITE_VERIFY.8.GZ/SG_WRITE_VERIFY.8","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00",9386,ALLOCATED,ALLOCATED,UNKNOWN,"A8C1BBFFE58D1D18BFEC08D3F1D8AC41","0ADC7366E07492747A043E02DC71321903873ABD29FE31E3A34424F054A78569",TEXT/PLAIN,8

CH9.H,"PTOR, OR

**
* (RARELY) Â«ACCEPTEDÂ« BY SET_DESCRIPTOR.

*","/IMG_ELK_VAR_PARTICION.DD/USR/INCLUDE/LINUX/USB/CH9.H","2025-10-10 18:38:46 ECT","2025-11-28 18:20:07 ECT","2025-11-28 18:20:05 ECT","2025-11-28 18:20:07 ECT",39930,ALLOCATED,ALLOCATED,UNKNOWN,"A7FC85D0AB2751EC11E5BB7F6F4DE974","5CF14885693F7D64617864E1396D1929631441CC382FFFCEB647DA1A40B6E1E3",TEXT/X-CHDR,H

C_CPP.PY,"ADDITIONAL OPTIONS Â«ACCEPTEDÂ«:

`STDLIBHIGHLIGHTING`","/IMG_ELK_VAR_PARTICION.DD/USR/LIB/PYTHON3/DIST-PACKAGES/PYGMENTS/LEXERS/C_CPP.PY","2024-02-02 08:26:20 ECT","2025-11-28 18:17:39 ECT","2025-11-28 18:17:39 ECT","2025-11-28 18:17:39 ECT",17946,ALLOCATED,ALLOCATED,UNKNOWN,"1A54F4799ADB674A9327AD9F5BA5EBA9","D7B3BBB597E70C842A8E71446A31D572AA167F70A93FC2E85A5076C7C6AFC061",TEXT/X-PYTHON,PY

WAITER.CPYTHON-312.PYC,"EXCEEDED. PREVIOUSLY Â«ACCEPTEDÂ« STATE: %SR8

POPR,

LOGGER","/IMG_ELK_VAR_PARTICION.DD/USR/LIB/PYTHON3/DIST-PACKAGES/BOTOCORE/__PYCACHE__/WAITER.CPYTHON-312.PYC","2025-08-05 12:14:02 ECT","2025-11-28 18:17:35 ECT","2025-11-28 18:17:35 ECT","2025-11-28 18:17:35 ECT",14580,ALLOCATED,ALLOCATED,UNKNOWN,"96E25903A3BD0F5E0738D040DD8815D5","95ADDAC46B3AE47EEC18EA070443A5F71FC40FB79286D8C60164A4EB4016CB30",APPLICATION/OCTET-STREAM,PYC

XMLPARSER.JS,"NEW ERROR(""XML DATA IS Â«ACCEPTEDÂ« IN STRING OR BYTES[]","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/KIBANA/NODE_MODULES/FAST-XML-PARSER/SRC/XMLPARSER/XMLPARSER.JS","2025-11-07 08:37:48 ECT","2025-11-28 18:55:04 ECT","2025-12-01 07:41:35 ECT","2025-11-28 18:55:02 ECT",2120,ALLOCATED,ALLOCATED,UNKNOWN,"2D8E32340F186E976069A34F8FFEE32B","EC56754198D84D73CCEFF5247566FB861E9D1939185FD96EB83A115C9E67F886",TEXT/JAVASCRIPT,JS	
PASCAL.PY,"ADDITIONAL OPTIONS Â«ACCEPTEDÂ«: `TURBOPASCAL`","/IMG_ELK_VAR_PARTICION.DD/USR/LIB/PYTHON3/DIST-PACKAGES/PYGMENTS/LEXERS/PASCAL.PY","2024-02-02 08:26:20 ECT","2025-11-28 18:17:39 ECT","2025-11-28 18:17:39 ECT","2025-11-28 18:17:39 ECT",30880,ALLOCATED,ALLOCATED,UNKNOWN,"7F2B4810C8048DFDAB3B7B50F19457D9","107662FE687857B2EE197B5C6D3427AEB7EA0C35DC1B25700EF26637C5D1729B",TEXT/X-PYTHON,PY	
LVCHANGE.8,"SPECIFIC TYPE, WHERE THE Â«ACCEPTEDÂ« LV TYPES ARE LISTED.","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/MAN/MAN8/LVCHANGE.8.GZ/LVCHANGE.8","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00",30971,ALLOCATED,ALLOCATED,UNKNOWN,"3134E5A49F61053358C6DF128BC8DFBD","5373BAD5F95442CB853D62907ACEE297F8C59C4085B6163D662BAD9BD5F72C53",TEXT/PLAIN,8	
ENDPOINTS.PY,"THE CERTIFICATE) ARE Â«ACCEPTEDÂ« AND USED TO CONSTRUCT","/IMG_ELK_VAR_PARTICION.DD/USR/LIB/PYTHON3/DIST-PACKAGES/TWISTED/INTERNET/ENDPOINTS.PY","2025-08-05 12:14:03 ECT","2025-11-28 18:17:40 ECT","2025-11-28 18:17:40 ECT","2025-11-28 18:17:40 ECT",78428,ALLOCATED,ALLOCATED,UNKNOWN,"276437EFBE943CA6C63A44C147F6C9F0","C30A317D3137AF0C736C121EB7B008CEA75B76F4BC0C95D840B135359484029D",TEXT/X-PYTHON,PY	
JOBSTATEREASON.JAVA,"WHEN THE JOB	

```
    * IS Â«ACCEPTEDÂ«, OR
SUBSEQUENTLY
WHILE","/IMG_ELK_VAR_PARTICION.DD/US
R/SHARE/LOGSTASH/JDK/LIB/SRC.ZIP/JAVA
.DESKTOP/JAVAX/PRINT/ATTRIBUTE/STAND
ARD/JOBSTATEREASON.JAVA","2025-10-21
00:00:00 ECT","0000-00-00 00:00:00","0000-
00-00 00:00:00","0000-00-00
00:00:00",19987,ALLOCATED,ALLOCATED,U
NKNOWN,"BD8E2A09CBE324969F1A5D1193
3AD1FA","1BE636D2ADCD94E02F953F4DEB
F018B577DA40A192D78630C972B7ED08651E
80",TEXT/X-JAVA-SOURCE,JAVA


LEXGROG.1,"THE ABOVE SHOULD BE
Â«ACCEPTEDÂ«.
.SH É–¢É€£É …Ç›®
.IF
!'PO4A'HIDE'","/IMG_ELK_VAR_PARTICION.
DD/USR/SHARE/MAN/JA/MAN1/LEXGROG.1.
GZ/LEXGROG.1","0000-00-00
00:00:00","0000-00-00 00:00:00","0000-00-00
00:00:00","0000-00-00
00:00:00",6553,ALLOCATED,ALLOCATED,UN
KNOWN,"08F31897DC67D0E32B8AC06935CA
D040","2D19FB410602C07F1ABBF439303DC
20D06DE28F7A09EF00DC99AF4853DB28D75
",TEXT/PLAIN,1
METHODGENERATOR.JAVA,"CHUNKS
WHICH HAVE BEEN Â«ACCEPTEDÂ« AS
CANDIDATES
CHUNKS","/IMG_ELK_VAR_PARTICION.DD/U
SR/SHARE/LOGSTASH/JDK/LIB/SRC.ZIP/JAV
A.XML/COM/SUN/ORG/APACHE/XALAN/INTE
RNAL/XSLTC/COMPILER/UTIL/METHODGEN
ERATOR.JAVA","2025-10-21 00:00:00
ECT","0000-00-00 00:00:00","0000-00-00
00:00:00","0000-00-00
00:00:00",95062,ALLOCATED,ALLOCATED,U
NKNOWN,"622BC248ED0F040101B20EE120E
F3BC1","CBD3C9EF6333C84BD539D2F00E87
1D3827F193AE457EE14B6AA7F648C12E4C94
",TEXT/X-JAVA-SOURCE,JAVA
```

FONT-FORMATS-4.HTML,"MULTIPLE OF 256 PLUS 40 ARE Â«ACCEPTEDÂ« AS POSSIBLE BINARY FONT","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/DOC/KBD/FONT-FORMATS/FONT-FORMATS-4.HTML","2025-08-05 12:14:04 ECT","2025-11-28 18:17:54 ECT","2025-11-28 18:17:54 ECT","2025-11-28 18:17:54 ECT",2198,ALLOCATED,ALLOCATED,UNKNOWN,"F463620EDA46EA23B741D74D00B3CD2D","D843B8111C23F76B4B8F3ADB4C7AFB25CB66B004D9B7BD6F05A45A30041FC73C",TEXT/HTML,HTML	
IUCODE_TOOL.8,"\FIMODE\FP ARGUMENT (Â«ACCEPTEDÂ« ONLY BY THE LONG VERSION","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/MAN/MAN8/IUCODE_TOOL.8.GZ/IUCODE_TOOL.8","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00",20671,ALLOCATED,ALLOCATED,UNKNOWN,"D010F2B0E63E15554E7B79FC62E5885A","18E0BC9AD9E487868D290716D36B232799C565DFC5B19B407AC1DD4A04FCEB6B",TEXT/TROFF,8	
CMP.H,"::= INTEGER {	
* Â«ACCEPTEDÂ« (0),","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/KIBANA/NODE/GLIBC-217/INCLUDE/NODE/OPENSSL/ARCHS/VC-WIN32/ASM_AVX2/INCLUDE/OPENSSL/CMP.H","2025-11-07 08:39:12 ECT","2025-11-28 18:55:03 ECT","2025-11-28 18:54:38 ECT","2025-11-28 18:54:39 ECT",41123,ALLOCATED,ALLOCATED,UNKNOWN,"0004A88D9AC2F2897FE756B3780DF38F","28ABBCA1B6798A226A323C5B23DD884D3FE739F5D9C8B0F1329CC665355D6317",TEXT/X-CHDR,H	
INTERFACES.PY,"TO HANDLE CONNECTIONS Â«ACCEPTEDÂ« VIA THIS SOCKET.","/IMG_ELK_VAR_PARTICION.DD/USR/LIB/PYTHON3/DIST-PACKAGES/TWISTED/INTERNET/INTERFACES.PY","2025-08-05 12:14:03 ECT","2025-11-28 18:17:40 ECT","2025-11-28 18:22:26 ECT","2025-11-28 18:17:40 ECT",98187,ALLOCATED,ALLOCATED,UNKNOWN,"8B6D7FF0E8293ECA458043287FC84610","5EBCB9B7103EFFE780EBBC7F69F09DE5E1214AEA119BCD95A3BA1481B232580A",TEXT/X-PYTHON,PY	

PARSEEXCEPTION.JAVA,"THE MESSAGE SUPPLIER IS Â«ACCEPTEDÂ« NOT TO CONTROL WHEN TO","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/LOGSTASH/JDK/LIB/SRC.ZIP/JDK.JAVADOC/JDK/JAVADOC/INTERNAL/DOCLETS/TOOLKIT/TAGLETS/SNIPPET/PARSEEXCEPTION.JAVA","2025-10-21 00:00:00 ECT","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00",2054,ALLOCATED,ALLOCATED,UNKNOWN,"22C807C6DBC94FE2326B6842BF06DFEB","7BD0DA1043E9F23888FD9F9E3CBDE3720B8C2C8AF93357E135378CC29ADC1CC8",TEXT/X-JAVA-SOURCE,JAVA	
DPKG-DISTADDFILE.1,"18.5). THE CURRENTLY Â«ACCEPTEDÂ« VALUES ARE: \FBAUTO\FR","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/MAN/JA/MAN1/DPKG-DISTADDFILE.1.GZ/DPKG-DISTADDFILE.1","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00",3941,ALLOCATED,ALLOCATED,UNKNOWN,"85B6364F6706F63536E4B6355AA8068C","B7CF8FCD8E2A1F67B7421EAFDD636D08D7775C62AE1887583D5F5C193628F3B2",TEXT/TROFF,1	
X509CRLSELECTOR.JAVA,"THE MINIMUM CRL NUMBER Â«ACCEPTEDÂ« (OR {@CODE NULL})","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/LOGSTASH/JDK/LIB/SRC.ZIP/JAVA.BASE/JAVA/SECURITY/CERT/X509CRLSELECTOR.JAVA","2025-10-21 00:00:00 ECT","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00",28097,ALLOCATED,ALLOCATED,UNKNOWN,"5DE22075CA20BE9DF21D4B5B428C1450","98BC3350E7115FBC5E176D1427F20EDEAE998AC03A01100B1868ED708C383580",TEXT/X-JAVA-SOURCE,JAVA	
DIRMNGR.8,"AND THEM LOCAL	

**CLOCK IS Â«ACCEPTEDÂ«. DEFAULT IS 600
(10","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/MAN/MAN8/DIRMNGR.8.GZ/DIRMNGR.8","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00",29091,ALLOCATED,ALLOCATED,UNKNOWN,"877CD4151D1CDC179E7382D7B7A99383","B271ED34F0AB2D145252FAC71BBD26E07CBCBD3A2338BF5E09FBEF33270D57FB",TEXT/TROFF,8**

X509CERTSELECTOR.JAVA,"* CERTIFICATES ARE Â«ACCEPTEDÂ«. IF THE VALUE IS -1,","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/LOGSTASH/JDK/LIB/SRC.ZIP/JAVA.BASE/JAVA/SECURITY/CERT/X509CERTSELECTOR.JAVA","2025-10-21 00:00:00 ECT","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00",104021,ALLOCATED,ALLOCATED,UNKNOWN,"F54C13F69041E6789F549D062B0D0F49","0B82BA5C54949C9090BE137F6131F0A5DF6EFE897AC87CB5608F5C560C891210",TEXT/X-JAVA-SOURCE,JAVA

SG_RESET.8,"\FI\-\-NO-ESCALATE\FR IS ALSO Â«ACCEPTEDÂ«.

.TP

\FB\-\-NO\-ESCALATE\FR","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/MAN/MAN8/SG_RESET.8.GZ/SG_RESET.8","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00",6395,ALLOCATED,ALLOCATED,UNKNOWN,"809893C81C14DB71DB61B7EC83CF09E1","D348F1259F53E3E158D166F8BF0F5518C5BF3AAABADABAA6A9E73E2BB3095F92",TEXT/PLAIN,8

LVRESIZE.8,"SPECIFIC TYPE, WHERE THE Â«ACCEPTEDÂ« LV TYPES ARE LISTED.","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/MAN/MAN8/LVRESIZE.8.GZ/LVRESIZE.8","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00",17667,ALLOCATED,ALLOCATED,UNKNOWN,"62C4591DEA568D50C92ECE8964B4BC53","F1AADB0E9C492CBEEBBFDEFB072A0E0A57992C4BD3B72912DFDEE466715B920D",TEXT/PLAIN,8

NUMBEROFDOCUMENTS.JAVA,"DOCS THE PRINTER HAS Â«ACCEPTEDÂ« FOR THIS

* JOB, REGARDLESS","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/LOGSTASH/JDK/LIB/SRC.ZIP/JAVA.DESKTOP/JAVAX/PRINT/ATTRIBUTE/STANDARD/NUMBEROFDOCUMENTS.JAVA","2025-10-21 00:00:00 ECT","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00",4052,ALLOCATED,ALLOCATED,UNKNOWN,"7D49CC794198C4931B6A8981144E8205","13CE23DF2120AA6ABCA0B6FCB05493279C5CF269EDF4402EA1C9E6EEF8D3E8F9",TEXT/X-JAVA-SOURCE,JAVA	
PHP.PY,"ADDITIONAL OPTIONS Â«ACCEPTEDÂ«: `STARTINLINE`	
Â€ Â€ Â€ ","/IMG_ELK_VAR_PARTICION.DD/USR/LIB/PYTHON3/DIST-PACKAGES/PYGMENTS/LEXERS/PHP.PY","2024-02-02 08:26:20 ECT","2025-11-28 18:17:39 ECT","2025-11-28 18:17:39 ECT","2025-11-28 18:17:39 ECT",13040,ALLOCATED,ALLOCATED,UNKNOWN,"6275B0EE3071385EF2C2E10D5BA49727","6F8B155994CCAB4B7618C95A166CD85899DAF56B83405226FAF76A5E46FCF71A",TEXT/X-PYTHON,PY	
MANPATH.5,"""").	
\FBSECTIONS\FP IS Â«ACCEPTEDÂ« AS AN ALTERNATIVE NAME","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/MAN/JA/MAN5/MANPATH.5.GZ/MANPATH.5","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00",7604,ALLOCATED,ALLOCATED,UNKNOWN,"9EB542B5E2213979E5AC3763F48B71E9","C45EFA338EE4AD190F68419211688D016FF560FBDC89E81D9A1C102ED0EFD8E4",TEXT/TROFF,5	

SERVICE-2.JSON,"THE LIST OF OPEN AND Â«ACCEPTEDÂ« BEHAVIOR GRAPH INVITATIONS","/IMG_ELK_VAR_PARTICION.DD/USR/LIB/PYTHON3/DIST-PACKAGES/BOTOCORE/DATA/DETECTIVE/2018-10-26/SERVICE-2.JSON","2025-08-05 12:14:02 ECT","2025-11-28 18:17:36 ECT","2025-11-28 18:17:36 ECT","2025-11-28 18:17:36 ECT",93497,ALLOCATED,ALLOCATED,UNKNOWN,"5082F7DFF52C92CBE2B466EAD75243DC","A319A158A7AD308AADDA61BF07495BAE1CCC07D7290756BFA7C67946DD032ED7",TEXT/PLAIN,JSON
SCRIPTING.PY,"ADDITIONAL OPTIONS Â«ACCEPTEDÂ«:

`FUNC_NAME_HIGHLIGHTING`","/IMG_ELK_VAR_PARTICION.DD/USR/LIB/PYTHON3/DIST-PACKAGES/PYGMENTS/LEXERS/SCRIPTING.PY","2024-02-02 08:26:20 ECT","2025-11-28 18:17:39 ECT","2025-11-28 18:17:39 ECT","2025-11-28 18:17:39 ECT",70014,ALLOCATED,ALLOCATED,UNKNOWN,"F421583DD1DE27BF56C698DE6EC21F7B","1245CACD62E0FA3A6C57CADEA76616E9F0EC083CFE7777D8D6ABF541513B308F",TEXT/X-PYTHON,PY
TCP.PY,"WHEN A CONNECTION IS Â«ACCEPTEDÂ«, THIS WILL CALL A FACTORY'S","/IMG_ELK_VAR_PARTICION.DD/USR/LIB/PYTHON3/DIST-PACKAGES/TWISTED/INTERNET/TCP.PY","2025-08-05 12:14:03 ECT","2025-11-28 18:17:40 ECT","2025-11-28 18:22:26 ECT","2025-11-28 18:17:40 ECT",55269,ALLOCATED,ALLOCATED,UNKNOWN,"DAADC92B76D3C71C598824AAF04C96DC","F96FDD9B6C5F2727762E728123A494460E777841784D4850A267412208B13BE1",TEXT/X-PYTHON,PY
CLIENT.RB,"LIST OF THE

# Â«ACCEPTEDÂ« MODES AND THE CONFIGURATION","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/LOGSTASH/VENDOR/BUNDLE/JRUBY/3.1.0/GEMS/AWS-SDK-CORE-3.226.3/LIB/AWS-SDK-SSO/CLIENT.RB","2025-11-04 13:09:44 ECT","2025-11-28 21:59:32 ECT","2025-11-28 21:59:24 ECT","2025-11-28 21:59:25 ECT",32190,ALLOCATED,ALLOCATED,UNKNOWN,"36AAD5E3998144532C09878A95D92A77","F5BBDBAAC8146EB423C7CCF37EBBAFB5708D8BCA624A3AB47B6028F7D3DAF082",TEXT/X-RUBY,RB	
TIPC_CONFIG.H.DPKG-NEW,"ISSUED BY ANY PROCESS.	
*** Â«ACCEPTEDÂ« BY OWN NODE, OR BY REMOTE","/IMG_ELK_VAR_PARTICION.DD/USR/INCLUDE/LINUX/TIPC_CONFIG.H.DPKG-NEW","2025-10-10 18:38:46 ECT","2025-11-28 18:20:07 ECT","2025-11-28 18:20:05 ECT","2025-11-28 18:20:07 ECT",14915,UNALLOCATED,ALLOCATED,UNKNOWN,"CE64970E74DA3B78B11F97A3C7AA5E84","010BDA2F00ED4E9E26CA567B0306CDBF5BABDDB1855C2AA8CAF1D2841A7702BB",TEXT/PLAIN,DPKG-NEW**	
X509_VFY.H,"TRUST IF NO EXPLICIT Â«ACCEPTEDÂ« TRUST EKUS */	
# DEFINE","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/KIBANA/NODE/GLIBC-217/INCLUDE/NODE/OPENSSL/ARCHS/VC-WIN32/ASM_AVX2/INCLUDE/OPENSSL/X509_VFY.H","2025-11-07 08:39:12 ECT","2025-11-28 18:55:03 ECT","2025-11-28 18:54:38 ECT","2025-11-28 18:54:39 ECT",52027,ALLOCATED,ALLOCATED,UNKNOWN,"03954943980620D1194BEB68249B9EC7","00ECA88DA5CAC83641BA85391C7351252411AF33B9F9BF12A31CDA3D10F5C2F1",TEXT/X-CHDR,H	

KCONFIG,"SECURITY MODELS (LSMS) Â«ACCEPTEDÂ« BY APPARMOR, SELINUX","/IMG_ELK_VAR_PARTICION.DD/USR/SRC/LINUX-HEADERS-6.8.0-88/FS/UBIFS/KCONFIG","2024-03-10 15:38:09 ECT","2025-11-28 18:20:17 ECT","2025-11-28 18:20:05 ECT","2025-11-28 18:20:15 ECT",3328,ALLOCATED,ALLOCATED,UNKNOWN,"70D54F8A07E2A25921D84416791E8598","18978925EEAFCDC274BE44F0A368144F29CB89CF1A190815E4B8836F27B69243",TEXT/PLAIN,

MIDISYSTEM.JAVA,"THE RESPECTIVE TYPE IS Â«ACCEPTEDÂ«, TOO
IF ("/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/LOGSTASH/JDK/LIB/SRC.ZIP/JAVA.DESKTOP/JAVAX/SOUND/MIDI/MIDISYSTEM.JAVA","2025-10-21 00:00:00 ECT","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00",65761,ALLOCATED,ALLOCATED,UNKNOWN,"31B6CE07903CE172A62A00B89D2C04D0","83BEEF97701362381B0FADFF22E55843B5233ECFCEE7175F5CF753FAA7F723AA",TEXT/X-JAVA-SOURCE,JAVA

CORESIGHT-STM.H,"ENSURE THE TRANSACTION IS Â«ACCEPTEDÂ« BY THE STM. WHILE INVARIANT","/IMG_ELK_VAR_PARTICION.DD/USR/INCLUDE/LINUX/CORESIGHT-STM.H","2025-10-10 18:38:46 ECT","2025-11-28 18:20:07 ECT","2025-11-28 18:20:05 ECT","2025-11-28 18:20:07 ECT",747,ALLOCATED,ALLOCATED,UNKNOWN,"61DC10BB4998AD40F65EA9B8C7F8ECCA","81CA4E16A8D91B490D824277CBDCD73394DC07CE77688B07ECF0C0D86D7165A0",TEXT/X-CHDR,H

REASONS.RB,"""CREATED"",
202 => ""Â«ACCEPTEDÂ«"",
Â€ Â€ Â€ (Ä¸€)ÃƑ”Ã‚³

=> ""NON","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/LOGSTASH/VENDOR/BUNDLE/JRUBY/3.1.0/GEMS/HTTP-3.3.0/LIB/HTTP/RESPONSE/STATUS/REASONS.RB","2025-11-04 13:09:44 ECT","2025-11-28 21:59:32 ECT","2025-11-28 21:59:24 ECT","2025-11-28 21:59:28 ECT",2476,ALLOCATED,ALLOCATED,UNKNOWN,"912C4BC21DEA3F501ABA1FA7EDFADFCC","2600434C4ACF5CA5BE844C49A58635D92890FF35EB221288104BBD5F2C4A8110",TEXT/X-RUBY,RB	
START.JAVA,"// CHECK IF ARG IS Â«ACCEPTEDÂ« BY THE TOOL BEFORE EMITTING","/IMG_ELK_VAR_PARTICION.DD/USR/SHARE/LOGSTASH/JDK/LIB/SRC.ZIP/JDK.JAVADOC/JDK/JAVADOC/INTERNAL/TOOL/START.JAVA","2025-10-21 00:00:00 ECT","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00",34861,ALLOCATED,ALLOCATED,UNKNOWN,"99AAA69B8789DA99CDC7CB2821A72570","231B93F1BE83EE0BAB101D33F97128581498C5146091AA024B6EACC17B778149",TEXT/X-JAVA-SOURCE,JAVA	
METRICBEAT,"COMCONFIG DOES NOT MATCH Â«ACCEPTEDÂ« CONFIGURATIONSCONFIGURATION","/IMG_ELK_VAR_PARTICION.DD/HOME/ELK/ELASTIC-AGENT-8.11.1-LINUX-X86_64.TAR.GZ/ELASTIC-AGENT-8.11.1-LINUX-X86_64.TAR/ELASTIC-AGENT-8.11.1-LINUX-X86_64/DATA/ELASTIC-AGENT-03EF9D/COMPONENTS/METRICBEAT","2023-11-11 08:27:08 ECT","0000-00-00 00:00:00","0000-00-00 00:00:00","0000-00-00 00:00:00",190722520,ALLOCATED,ALLOCATED,UNKNOWN,"06514FFF08CB116F92F91178A2AAE1E9","7E1C400E8BEB7529BD23612C8CC8DF084BCADE48F93E12CB267F4B45297B1069",APPLICATION/X-SHAREDLIB,	

Archivo _pr_es812postings_0.tim encontramos artefactos de indicios de ataques de fuerza bruta

Hacemos un análisis detallado: parece ser un fragmento de logs del sistema y de Logstash, mezclados con logs de PAM y algunos errores de red o conexión

Logs de Logstash

```
/var/log/logstash/logstash-plain-2025-11-30-9.log.gz
/var/log/logstash/logstash-plain-2025-12-01-1.log.gz
...
error: Protocol major versions differ: 2 vs. 1
```

Los archivos .log.gz son logs comprimidos de Logstash, probablemente rotados diariamente o por tamaño.

El mensaje Protocol major versions differ: 2 vs. 1 indica problemas de compatibilidad de versión de protocolo, generalmente SSH o de un input/output de Logstash que intenta comunicarse con otro servicio (por ejemplo, beats o Elasticsearch).

MaxStartups throttling

```
beginning MaxStartups throttling
exited MaxStartups throttling after 00:00:01, 2 connections dropped
...
exited MaxStartups throttling after 00:06:57, 808 connections dropped
```

Esto es el mensaje típico de OpenSSH.

MaxStartups define cuántas conexiones SSH simultáneas puede aceptar el servidor antes de empezar a rechazar nuevas conexiones.

El log indica que durante periodos específicos muchas conexiones fueron rechazadas:

Ej: 808 conexiones perdidas en ~7 minutos.

Esto podría indicar un ataque de fuerza bruta SSH o un tráfico muy alto legítimo.

Logs de PAM (autenticación)

```
pam_unix(cron:session): session opened/closed for user root
pam_unix(login:session): session opened for user elk
pam_unix(sshd:auth): authentication failure; rhost=192.168.100.15
user=atacanteg7
pam_unix(sshd:session): session opened for user atacanteg7
pam_unix(sudo:session): session opened for user root by elk
pam_unix se refiere al módulo de autenticación Unix/PAM.
```

Lo más relevante aquí es que Hay fallos de autenticación SSH desde la IP 192.168.100.15 con usuarios atacanteg7 y root

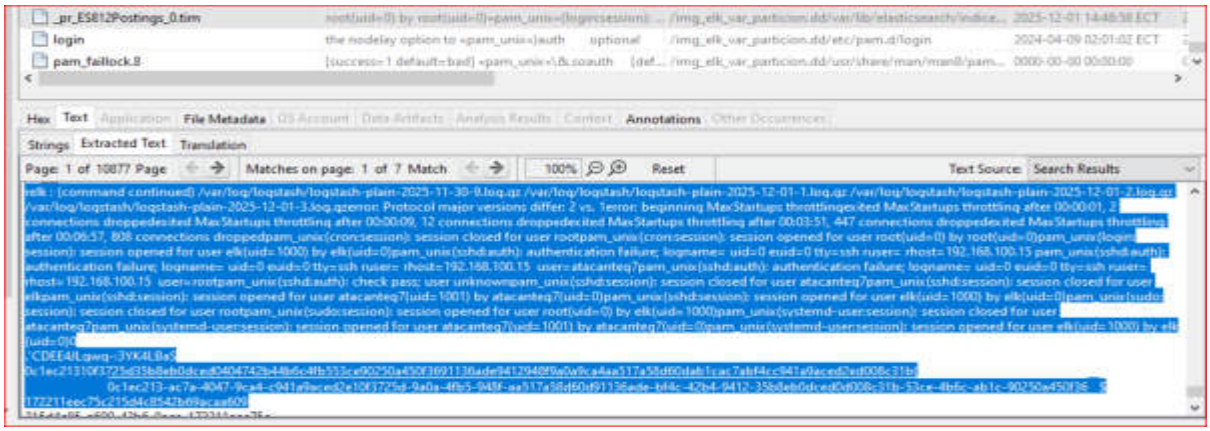
También hay sesiones válidas abiertas y cerradas para usuarios elk y atacanteg7.

Uso de sudo por el usuario elk.

Esto sugiere que alguien intentó conectarse por SSH con credenciales incorrectas, y algunas sesiones fueron creadas correctamente. Si atacanteg7 no es un usuario legítimo, esto indica un posible intento de intrusión.

Figura 163

Log obtenido con el usuario atacanteg7



Archivo auth.log y pipeline.js

Podemos determinar claramente que hay muchos intentos fallidos de inicios de sesión utilizando el comando sudo, esto nos da indicios de posibles ataques en la infraestructura debido a que el usuario de Ubuntu intenta logearse pero sin éxito.

Usuarios atacados

Se observa un patrón claro de prueba de múltiples nombres de usuario, tanto válidos como inválidos:

```
Invalid users (no existen en el sistema): atacante, grupo7, ubuntu
```

Usuarios válidos: elk

Patrón típico: un ataque de fuerza bruta dirigido a probar usuarios comunes o específicos del sistema.

Tipo de fallos

En los logs aparecen mensajes repetidos como:

```
pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0
tty=ssh ruser= rhost=192.168.100.15
```

Failed password for invalid user atacante from 192.168.100.15 port 59974 ssh2

Connection closed by invalid user atacante 192.168.100.15 port 59974 [preauth]

Interpretación:

Invalid user: El usuario no existe en el sistema.

authentication failure: La contraseña fue incorrecta.

Connection closed [preauth]: SSH cerró la conexión antes de autenticarse.

Se repite múltiples veces por usuario y puerto en segundos, mostrando automatización.

Patrón de ataque

Se observan características típicas de fuerza bruta SSH:

Alta frecuencia de intentos: múltiples conexiones por usuario en cuestión de segundos.

Puertos cambiantes: el atacante usa diferentes puertos locales en cada intento.

Usuarios múltiples: se prueba con usuarios comunes, invalidos y específicos (elk).

Cronología resumida:

Tabla 2

Cronología resumida

USUARIO	INTENTOS	ESTADO	COMENTARIO
ATACANTE	>10	Fallo	Usuario no existe
ELK	8-10	Fallo	Usuario válido, contraseña incorrecta
GRUPO7	6-8	Fallo	Usuario no existe
UBUNTU	6-8	Fallo	Usuario no existe

El ataque comienza alrededor de 11:28:57 y continua hasta 11:30:11.

Cada usuario recibe múltiples intentos en cuestión de segundos.

Impacto en la seguridad

Intentos fallidos: no se detectaron inicios de sesión exitosos.

Riesgos potenciales:

Si no se bloquea la IP, continuará intentando acceder.

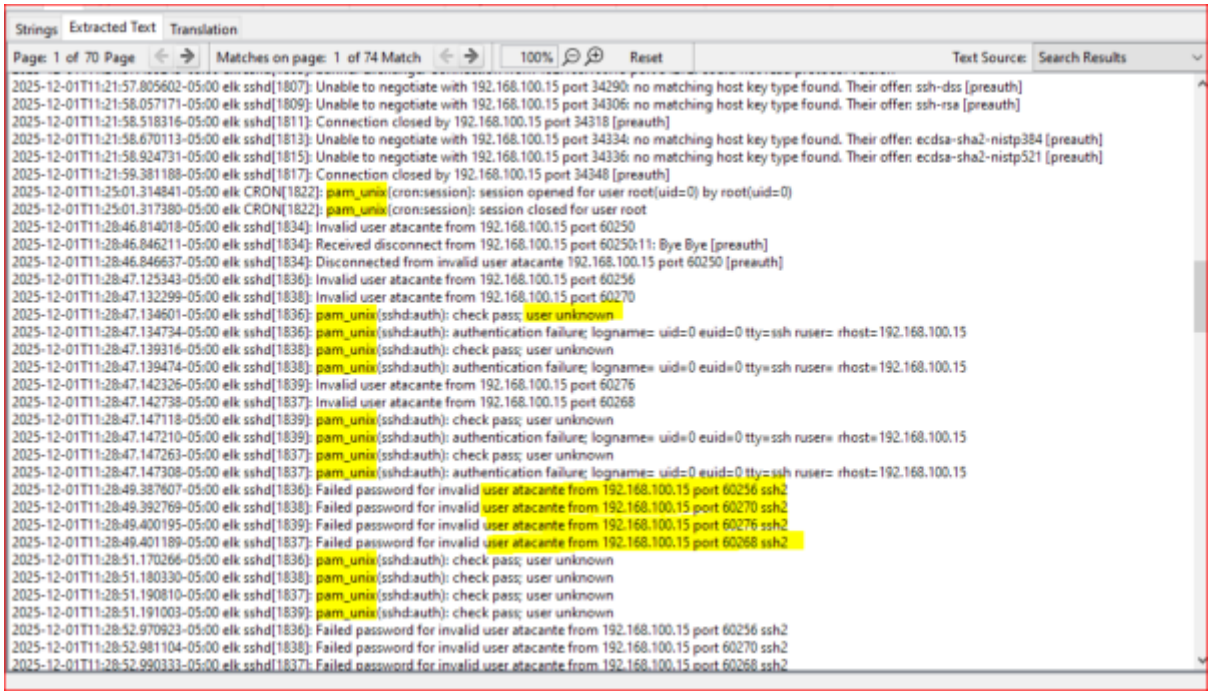
Puede saturar logs y alertas SIEM.

Intentos sobre usuarios válidos (elk) podrían dar acceso si la contraseña fuera débil.

auth.log

Figura 164

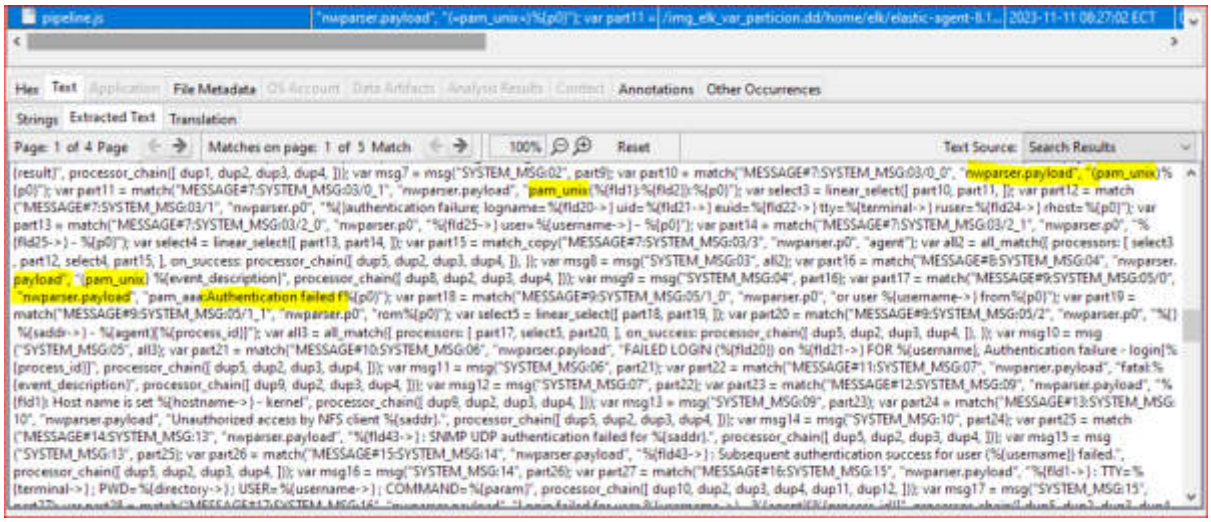
Captura del registro del archivo auth.log



pipeline.js

Figura 165

Salida de pipeline.js



Realizando el análisis respectivo y mediante la búsqueda con la ayuda del **keyword list** localizamos las credenciales he ingresamos a **/etc/Kibana** y encontramos que hay un riesgo crítico de seguridad:

Exposición de credenciales

```
elasticsearch.username: "kibana_system"
elasticsearch.password: "elkstackG7"
```

Esto significa que la contraseña de acceso a Elasticsearch está en texto plano dentro del archivo de configuración.

Si alguien tiene acceso a este archivo, puede conectarse a Elasticsearch con privilegios de sistema, potencialmente leer o modificar todos los índices.

Servidor expuesto a toda la red

```
server.host: 0.0.0.0
```

Esto hace que Kibana escuche en todas las interfaces de red, no solo localhost.

Combinado con la contraseña en texto plano, cualquiera que llegue al puerto 5601 podría intentar conectarse.

Esto es especialmente peligroso si la máquina tiene acceso público a Internet.

Otros puntos a revisar (menos críticos en este momento)

SSL está deshabilitado:

```
#server.ssl.enabled: false
```

Las conexiones son sin cifrado, por lo que las credenciales podrían interceptarse si alguien se conecta desde otra máquina en la red.

No hay configuración de firewall ni restricción de IP.

Este archivo sí contiene información sensible que podría ser explotada.

Si Kibana está accesible desde la red, un atacante podría usar:

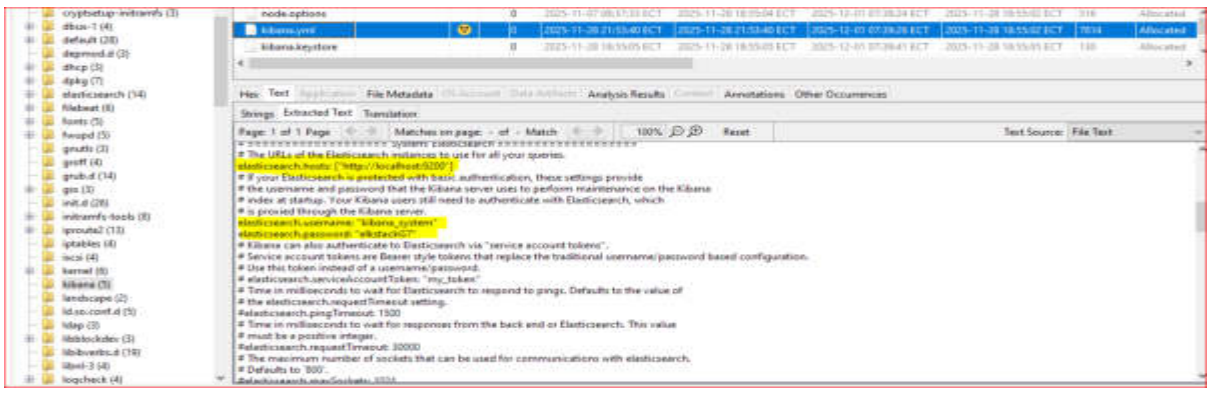
```
kibana_system / elkstackG7 para conectarse a Elasticsearch.
```

Leer, modificar o borrar índices.

Posible escalada dentro del stack ELK.

Figura 166

Evidencia de información sensible en archivo



Dentro de la estructura del análisis forense se encontraron evidencias que nos dan indicios claras y sospechosos de usuarios creados posiblemente sin autorización.

Buscamos en la ruta **/home/** encontramos dos usuarios creados en el SIEM **elk y atacanteg7**, el usuario principal y el usuario sospechoso.

Procedemos analizar el usuario creado posiblemente sin autorización.

La línea:

```
atacanteg7 ALL=(ALL) NOPASSWD: /bin/systemctl start kibana,
/bin/systemctl stop kibana, /bin/systemctl start elasticsearch,
/bin/systemctl stop elasticsearch
```

Nos indica:

Usuario al que aplica **atacanteg7**

Esta regla aplica solo al usuario **atacanteg7**.

Desde dónde puede usarla **ALL**

Este usuario puede ser ejecutado desde cualquier host.

Este usuario puede ejecutar los comandos **ALL**

Puede ejecutar los comandos como cualquier usuario, incluso como **root**:

```
sudo -u root ...
```

Aparentemente sin pedir contraseña **NOPASSWD**

Qué comandos específicos pueden ser ejecutados

```
/bin/systemctl start kibana
/bin/systemctl stop kibana
/bin/systemctl start elasticsearch
/bin/systemctl stop elasticsearch
```

Esto da permiso a **atacanteg7** para ejecutar comandos, por ejemplo:

```
sudo systemctl start kibana
sudo systemctl stop kibana
sudo systemctl start elasticsearch
sudo systemctl stop elasticsearch
```

Implicaciones de seguridad

Aunque no hay “claves expuestas”, esta regla sí:

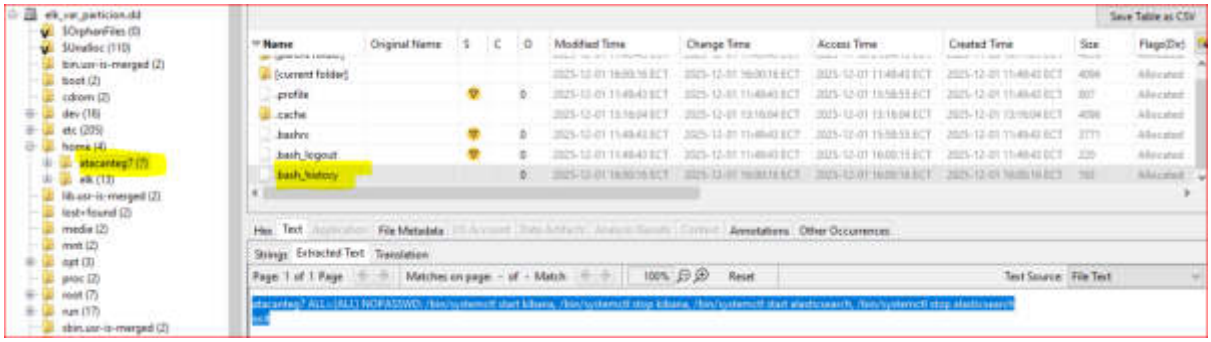
Permite a ese usuario controlar la disponibilidad y solo puede ejecutar exactamente esos comandos:

```
Kibana
Elasticsearch
```

No le da permisos completos de root, pero al detener Elasticsearch puede causar pérdida temporal de disponibilidad, interrumpir recolección de logs, detener aplicaciones dependientes, afectar a Kibana indirectamente.

Figura 167

Bash history en Elasticsearch



El usuario principal **/home/elk**, Procedemos analizar el usuario

En este usuario se puede analizar y evidenciar las vulnerabilidades expuestas como son:

Contraseñas expuestas del usuario elastic (Elasticsearch)

Se encontraron múltiples comandos donde se exponen las contraseñas:

Contraseña expuesta: **elkstackG7**

Aparecen en:

-E output.elasticsearch.password="elkstackG7"

```
curl -u elastic:elkstackG7 ...
```

```
curl -u elastic:elksatckG7 ...
```

Incluso aparece mal escrita en una línea del comando:

```
curl -u elastic:elksatckG7
```

Se encontraron Tokens de enrolamiento y service-tokens expuestos (Fleet / Elastic Agent)

Tokens expuestos:

Token 1 (enrollment token)

--enrollment-token

```
a0l3TDJwb0JQSWZiVk45RjROdDM6dVVkNWRtRC1GRTlNSVQ5VElzS1ZHUQ==
```

Token 2 (service token para Fleet)

--fleet-server-service-

```
token=a0l3TDJwb0JQSWZiVk45RjROdDM6dVVkNWRtRC1GRTlNSVQ5VElzS1ZHUQ==
```

Token 3 (otro service-token generado varias veces)

```
AAEAAWVsYXN0aWMvZmxlZXQtc2VydmVyL2ZsZWV0LTE6cVNZSEhRV1RRNDJDMmc1TjBfU
3BMdw
```

Enrollment tokens de Kibana

```
./bin/elasticsearch-create-enrollment-token -s kibana
```

Service tokens creados y listados en texto plano

```
/usr/share/elasticsearch/bin/elasticsearch-service-tokens create
elastic/fleet-server fleet-1
```

Tabla 3

Resumen de claves expuestas

TIPO	VALOR EXPUESTO	SE USA EN
CONTRASEÑA ELASTIC SUPERUSUARIO	elkstackG7	Curl, Filebeat, Logstash
VARIANTE MAL ESCRITA	elksatckG7	Curl
TOKENS DE ENROLAMIENTO (FLEET)	a0l3TDJwb0JQSWZi...	Elastic Agent
SERVICE TOKENS (FLEET SERVER)	AAEAAWVsYXN0aWMv...	Fleet Server
CONTRASEÑA PLACEHOLDER	TU_CLAVE	curl (si es real, está expuesta)

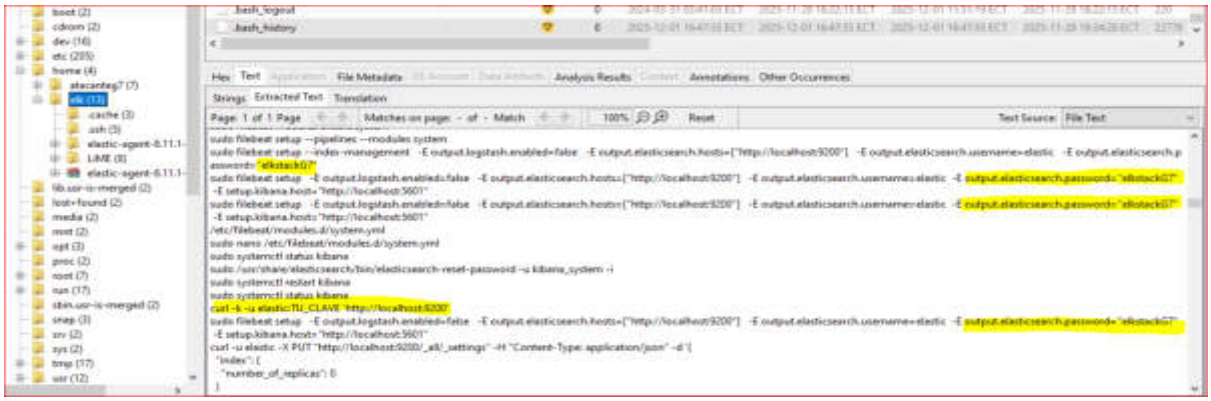
Impacto de seguridad

Estas credenciales permiten:

- ✓ Acceder a todo Elasticsearch con privilegios de superusuario.
- ✓ Registrar o manipular agentes en Fleet.
- ✓ Administrar logs, pipelines y datos.
- ✓ Potencialmente comprometer toda la infraestructura ELK.

Figura 168

Evidencia de impacto de seguridad



Encontramos evidencia de que se utilizó msfconsole, esto quiere decir que el SIEM estaba bajo ataques

ESTADO ACTUAL: ALTAMENTE SOSPECHOSO

Además del Metasploit Framework, el sistema tiene muchos paquetes Snap instalados, pero en particular solo uno es una herramienta de ataque: metasploit-framework.

Esto NO ocurre en un sistema limpio, ya que esto por sí solo indica que ya fuimos atacados, estos indicios se dan o nos indica que hubo actividad sospechosa fuerte, debido a que Metasploit no se instala solo.

Un atacante lo usa para:

- ✓ Escalar privilegios
- ✓ Explotar servicios locales
- ✓ Instalar backdoors
- ✓ Lanzar payloads
- ✓ Persistir

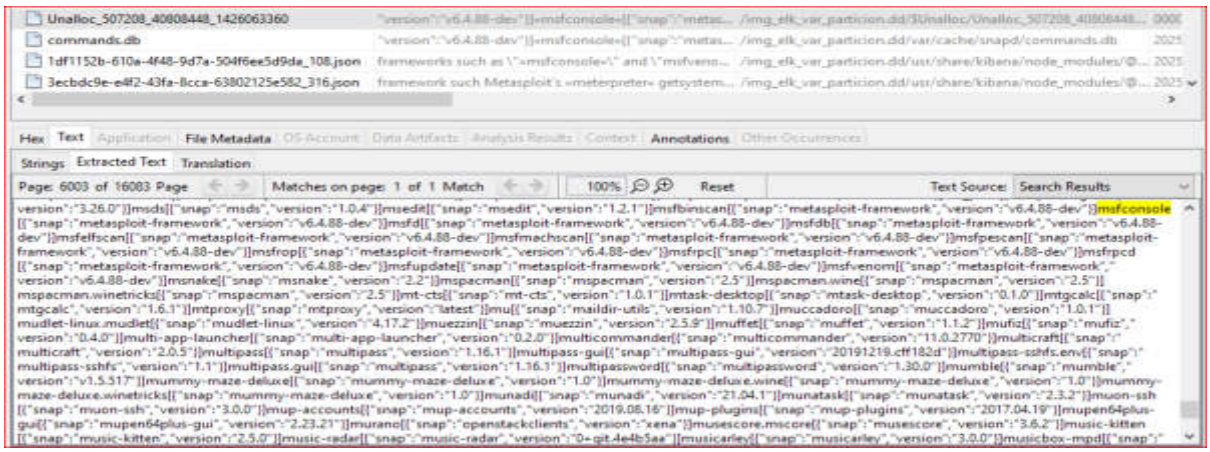
Metasploit Framework instalado esto NO es normal

Estos son los binarios peligrosos detectados:

- ✓ msfconsole
- ✓ msfvenom
- ✓ msfrpc
- ✓ msfrpcd
- ✓ msfbinscan
- ✓ msfd
- ✓ msfdb
- ✓ msfelfscan
- ✓ msfmachscan
- ✓ msfpescan
- ✓ msfrop
- ✓ msfupdate

Figura 169

Binarios peligrosos detectados



Exploramos más intensamente y detectamos **Meterpreter Reverse Shell**

Nombre: Potential Meterpreter Reverse Shell

Objetivo: Detectar actividad de Meterpreter en sistemas Linux. Meterpreter es un payload de Metasploit que permite a los atacantes ejecutar comandos y scripts de manera remota.

Comportamiento típico detectado:

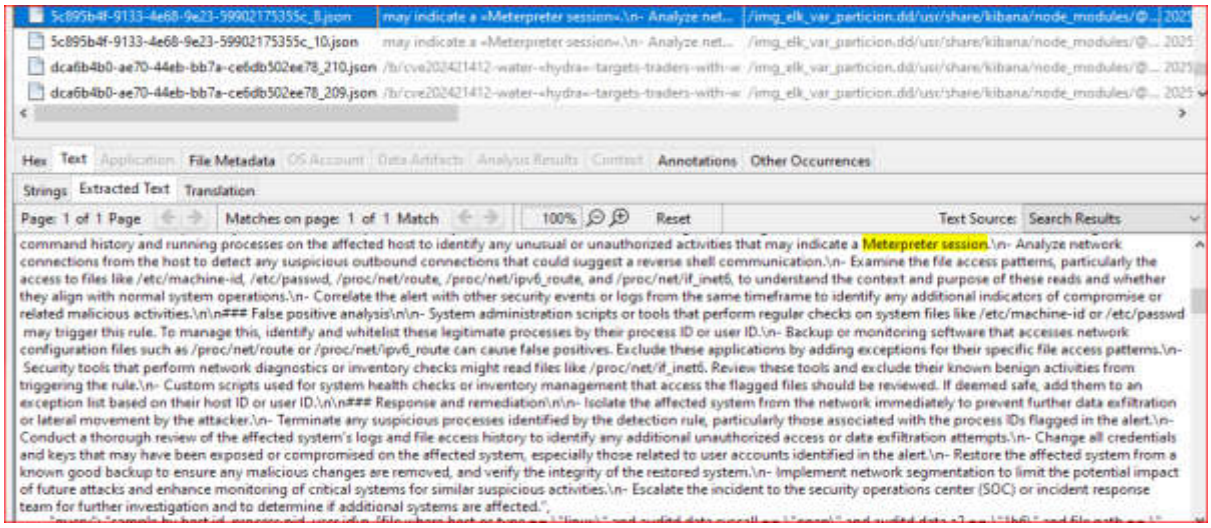
Acceso a archivos críticos de configuración y red, típicos de fingerprinting del sistema:

- ✓ /etc/machine-id
- ✓ /etc/passwd
- ✓ /proc/net/route
- ✓ /proc/net/ipv6_route
- ✓ /proc/net/if_inet6

Posible señal de un reverse shell activo o preparación de comandos remotos.

Figura 170

Posible señal de un reverse shell activo

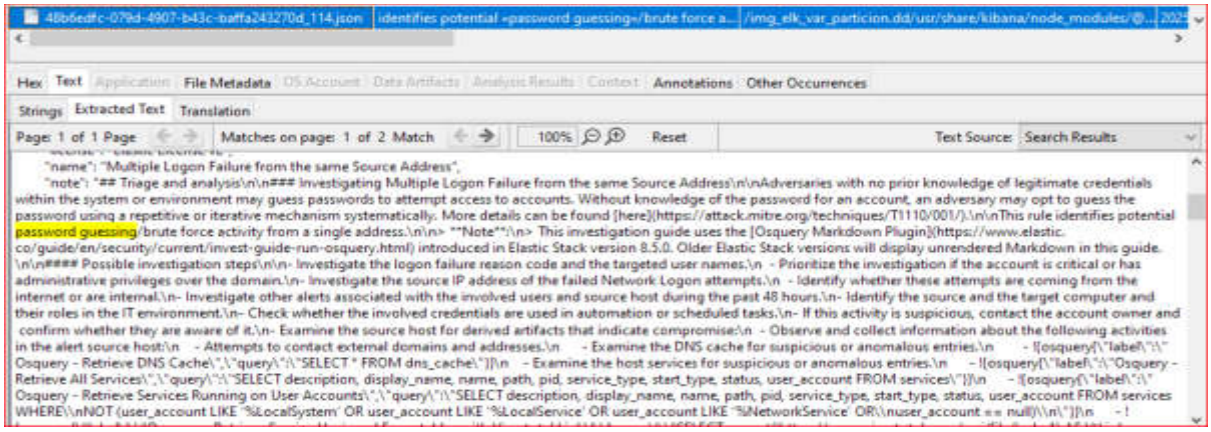


Este ataque describe una técnica de fuerza bruta de contraseñas, en la que el atacante intenta adivinar contraseñas mediante un método repetitivo o iterativo. En este caso, se enfoca en los intentos fallidos de inicio de sesión que provienen de una dirección IP en particular.

La técnica MITRE ATT&CK se utiliza para este ataque

Figura 171

Ataque con MITRE ATT&CK



5. Análisis Forense de la RAM memdump.lime con Volatility3

Primero ejecutamos el comando:

```
ython3 vol.py -f /home/kali/Documentos/memdump.lime linux.pslist
```

Verificamos el listado de procesos de la memoria LIME y extraemos la información del kernel y de los procesos activos.

Procesos Sospechosos Detectados

A continuación, enmarcamos solo los procesos que NO deberían estar normalmente en un sistema Linux limpio o que presentan indicios fuertes de manipulación, backdoor o rootkit:

Procesos psimon (PID 522 y 1423)

Psimon no es un proceso estándar de Linux, suele asociarse a herramientas corporativas de monitoreo agentes de telemetría no oficiales malware que registra actividad del sistema

Múltiples procesos Java sospechosos

Procesos:

PID 986 → java (UID 999/988)

PID 990 → java (UID 110)

PID 1612 → java (UID 110)

PID 1639 → controller (UID 110)

El proceso controller hijo de “java” es extremadamente raro.

Esto coincide con malware basado en:

web backdoors en Java

controladores remotos

cryptominers ocultos

C2 construido en Java

ALTAMENTE SOSPECHOSO.

Conexiones SSH anómalas

Procesos:

sshd (1854)
sesión remota sshd (2229)
login shell bash (2284)

Esto es normal si se conecta por SSH el personal autorizado.

Pero si NO es así puede ser acceso remoto no autorizado.

Procesos con EGID = 2147483648

Los procesos:

PID 3520

PID 3522

Tienen:

EGID = 2147483648 es una valor imposible, fuera de rango

Esto frecuentemente indica:

Manipulación de credenciales en memoria, intento de ocultar privilegios, rootkits que alteran estructuras de tareas, corrupción de memoria causada por un módulo malicioso cargado.

Esto es EXTREMADAMENTE SOSPECHOSO.

Uso de “insmod” (PID 3516)

PID 3516 → insmod

Este proceso carga módulos del kernel, si el personal autorizado no cargo un módulo manualmente en ese momento puede ser un rootkit o módulo malicioso.

Luego aparece muchísima actividad de kworker lo cual podría indicar un módulo alterado.

El sistema presenta varios indicadores de compromiso severo:

- Procesos no estándar (psimon)
- Múltiples procesos Java con usuarios extraños
- Proceso “controller” hijo de Java
- Sesiones SSH que podrían ser externas
- IDs imposibles (2147483648) → manipulación o rootkit
- “insmod” ejecutado justo antes de anomalías

- **Verificamos en el Anexo la Tabla 10**

Archivos ejecutados recientemente esto puede ser un indicador de malware.

Para ello ejecutamos el comando:

```
python3 vol.py -f /home/kali/Documentos/memdump.lime linux.bash
```

Esta ejecución nos indica que comandos fueron utilizados en qué fecha y a qué hora, los parámetros más destacados son:

Comandos relacionados con Elasticsearch

Incluyen:

Crear/borrar índices:

```
curl -X DELETE "localhost:9200/otherbeats-2025.11.29"
```

```
curl -X GET "localhost:9200/_cat/indices?v"
```

Modificación de templates:

```
"number_of_replicas": 0
"settings": { ... }
```

Cambios de permisos y directorios:

```
sudo chown -R elasticsearch:elasticsearch /var/lib/elasticsearch
sudo rm -rf /var/lib/elasticsearch/*
```

Reinicios:

```
sudo systemctl start|restart|stop elasticsearch
```

Cambios del archivo de configuración:

```
/etc/elasticsearch/elasticsearch.yml
```

Esto significa que se estuvo reinstalando, reseteando, configurando y depurando Elasticsearch.

Logstash

Incluye:

Crear/editar pipelines:

```
/etc/logstash/conf.d/02-beats-input.conf
/etc/logstash/conf.d/30-elasticsearch-output.conf
/usr/share/logstash/config/pipelines.yml
```

Validar configuración:

```
logstash -t
--config.test_and_exit
```

Permisos y directorios:

```
sudo chown logstash:logstash /var/lib/logstash
```

Escuchar puertos 514, 1514, 5044.

Claramente se realizó la configuración de Logstash como receptor de Syslog y Beats.

Kibana

Reinicios frecuentes:

```
sudo systemctl restart kibana
```

Cambios al archivo:

```
/etc/kibana/kibana.yml
```

Indicador de ajustes tras cambios en Elasticsearch y Filebeat.

Filebeat

Habilitar módulos:

```
sudo filebeat modules enable system
```

Setup de pipelines/índices/dashboards:

```
sudo filebeat setup
```

Reinicios:

```
sudo systemctl restart filebeat
```

Actividad normal cuando se instala Filebeat para syslogs.

Elastic Agent y Fleet

Comandos como:

Instalación:

```
sudo ./elastic-agent install
```

Uninstall:

```
sudo /opt/Elastic/Agent/elastic-agent uninstall
Tokens del Fleet Server:
```

```
elasticsearch-service-tokens create elastic/fleet-server
```

Significa que se intentó integrar el servidor con Fleet (Agente unificado de Elastic).

Firewall / Puertos

Aquí se encontró que hay muchos comandos:

Abrir puertos:

```
sudo ufw allow 514/udp
sudo ufw allow 1514/udp
sudo ufw allow 5044/tcp
sudo ufw allow 5601/tcp
sudo ufw allow 9200/tcp
```

Redirecciones con iptables:

```
iptables -t nat -A PREROUTING -p udp --dport 514 -j REDIRECT --to-
port 1514
```

Todo apunta a que se estaba configurando la recepción de Syslog en Logstash/Filebeat.

Limpieza de logs / espacio en disco

Muchos comandos peligrosos:

Borrados de logs:

```
sudo rm -rf /var/log/elasticsearch/*
sudo rm -rf /var/log/**/*.log
sudo truncate -s 0 /var/log/syslog
```

Limpiar journal:

```
sudo journalctl --vacuum-size=500M
sudo rm -rf /var/log/journal/*
```

Expandir LVM:

```
sudo lvextend -l +100%FREE ...
sudo resize2fs ...
```

Se incrementó el espacio en disco y se liberó espacio del sistema probablemente por un disco lleno que afectó Elasticsearch.

Configuración de SSH

Esto es muy llamativo:

```
sudo adduser atacanteg7
sudo sed -i 's/^#\?PasswordAuthentication.*/PasswordAuthentication
yes/' /etc/ssh/sshd_config
sudo systemctl restart ssh
```

Esto despierta alerta de seguridad, alguien creó un usuario llamado atacanteg7 y habilitó autenticación por contraseña.

Esto no es parte del mantenimiento de ELK.

Exploración de red

Nmap:

```
nmap --script ssh2-enum-algos -sV -p22 192.168.100.71
```

Esto evalúa algoritmos de cifrado SSH de servidores esto puede ser técnico de ser el caso autorizado si no es el caso esto es sospechoso.

Comandos de diagnóstico comunes

```
df -h
lsblk
pvs, vgs, lvs
netstat -tuln
```

Diagnósticos normales.

El proceso Bash ejecutó:

- Instalación, reconfiguración y debug intensivo de:
  - Elasticsearch
  - Logstash

- Kibana
- Filebeat
- Elastic Agent / Fleet

➢ Cambios en firewall, syslog y puertos

➢ Limpieza masiva de logs y expansión del LVM

➢ Muchas ediciones de configuración

Potenciales Indicios De Compromiso O Mal Uso

Los siguientes comandos no son normales:

➢ Creación del usuario:

- `adduser atacanteg7`

➢ Habilitar login por contraseña en SSH

- `PasswordAuthentication yes`

➢ Escaneo de servidores SSH con nmap

➢ Eliminación masiva de logs

- `rm -rf /var/log/*`

➢ Múltiples init 6 (reinicios)

Tabla 4

Potenciales indicios de compromiso o mal uso

TREE DEPTH	PID	PROCESS	COMMANDTIME	COMMAND
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo chown -R elasticsearch:elasticsearch /var/lib/elasticsearch
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo apt update
0	2284	bash	2025-12-01 21:11:16.000000 UTC	number_of_replicas: 0
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo apt install ufw
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo ufw allow 514/udp
0	2284	bash	2025-12-01 21:11:16.000000 UTC	otherbeats-2025.11.29
0	2284	bash	2025-12-01 21:11:16.000000 UTC	otherbeats-2025.11.30

0	2284	bash	2025-12-01 21:11:16.000000 UTC	curl -u elastic:elkstackG7 -X DELETE "localhost:9200/otherbeats-2025.11.29"
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo netstat -tuln | grep 514
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo netstat -tuln | grep 1514
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo ufw status numbered
0	2284	bash	2025-12-01 21:11:16.000000 UTC	}'
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo systemctl start filebeat
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo nano /etc/logstash/conf.d/02-beats-input.conf
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo /usr/share/logstash/bin/logstash -t
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo systemctl stop elasticsearch

0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo netstat -tuln | grep 514
0	2284	bash	2025-12-01 21:11:16.000000 UTC	}
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo iptables -t nat -A PREROUTING
				-p udp --dport 514 -j REDIRECT --to-port 1514
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo nano /etc/logstash/logstash.yml
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo apt update && sudo apt install curl
				apt-transport-https -y
0	2284	bash	2025-12-01 21:11:16.000000 UTC	df -h /
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo fdisk -l
0	2284	bash	2025-12-01 21:11:16.000000 UTC	lsblk

0	2284	bash	2025-12-01 21:11:16.000000 UTC	#!/bin/bash
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo systemctl restart kibana
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo rm -rf /var/log/*.log /var/log/*.gz /var/log/*.[0-9]
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo rm -rf /var/log/**/*.log /var/log/**/*.gz /var/log/**/*.[0-9]
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo systemctl status logstash
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo journalctl -u logstash -n 50 --no-pager
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo nano /etc/logstash/conf.d/02-beats-input.conf

0	2284	bash	2025-12-01 21:11:16.000000 UTC	init 6
0	2284	bash	2025-12-01 21:11:16.000000 UTC	[200~ls -l /usr/share/logstash/jdk/lib
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo netstat -tuln | grep 1514
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo nano /etc/elasticsearch/elasticsearch.yml
0	2284	bash	2025-12-01 21:11:16.000000 UTC	settings: {
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo apt update
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo filebeat setup –pipelines --modules system
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo filebeat modules enable system

0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo nano /etc/logstash/logstash.conf
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo netstat -tuln | grep -E '514|1514'
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo nano /etc/logstash/conf.d/
				02-beats-input.conf
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo rm -rf /var/log/elasticsearch/*
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo nano /etc/elasticsearch/
				elasticsearch.yml
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo lvextend -l +100%FREE
				/dev/mapper/ubuntu--vg-ubuntu--lv
0	2284	bash	2025-12-01 21:11:16.000000 UTC	Detectar filesystem y
				redimensionar automáticamente

0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo du -h --max-depth=1 /
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo du -h --max-depth=1 /var
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo rm -f /var/log/*.gz
0	2284	bash	2025-12-01 21:11:16.000000 UTC	}
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo nano /etc/filebeat/filebeat.yml
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo nano /etc/logstash/conf.d/
				30-elasticsearch-output.conf
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo -u logstash /usr/share/logstash/bin/logstash --config.test_and_exit -f /etc/logstash/conf.d/

0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo mkdir -p /var/lib/elasticsearch
0	2284	bash	2025-12-01 21:11:16.000000 UTC	export LD_LIBRARY_PATH=/usr/share/
				logstash/jdk/lib:$LD_LIBRARY_PATH
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo systemctl restart logstash
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo -u logstash /usr/share/logstash/bin/
				logstash --path.settings /etc/logstash -t
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo systemctl start elasticsearch
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo rm -rf /var/lib/elasticsearch/*
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo curl -u elastic -X PUT "http://localhost:9200/_index_template/
				default_template" -H "Content-Type:
				application/json" -d '{

0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo chmod -R 750 /var/log/elasticsearch
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo filebeat setup --index-management -E output.logstash.enabled=false -E output.elasticsearch.hosts=["http://localhost:9200"]
				-E output.elasticsearch.username=elastic -E output.elasticsearch.password="elkstackG7"
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo netstat -tuln | grep 1514
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo netstat -tuln | grep 514
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo nano /etc/logstash/conf.d/your-output-file.conf
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo filebeat modules list

0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo mkdir -p /var/log/elasticsearch
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo chown root:root /etc/elasticsearch/
				elasticsearch.yml
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo chmod -R 750 /var/lib/elasticsearch
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo systemctl restart logstash
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo systemctl status logstash
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo nano /etc/logstash/logstash.yml
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo netstat -tuln | grep 1514
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo netstat -tuln | grep 514

0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo systemctl status logstash
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo tail -f /var/log/logstash/logstash-plain.log
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo chown -R elasticsearch:elasticsearch /var/log/elasticsearch
0	2284	bash	2025-12-01 21:11:16.000000 UTC	GET /_search?q=*:*
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo ss -tulnp | grep 9200
0	2284	bash	2025-12-01 21:11:16.000000 UTC	template: {
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo rm -rf /etc/elasticsearch/*
0	2284	bash	2025-12-01 21:11:16.000000 UTC	curl -X GET "http://192.168.100.71:9200/_cat/indices?v"

0	2284	bash	2025-12-01 21:11:16.000000 UTC	@
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo chown logstash:logstash /var/lib/logstash
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo nano /etc/logstash/conf.d/ 01-syslog-input.conf
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo ls -l /var/log/logstash
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo init 0
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo apt install libwww-perl
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo journalctl -u logstash -f
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo ufw allow 514/tcp

0	2284	bash	2025-12-01 21:11:16.000000 UTC	[200~sudo systemctl restart logstash
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo nano /etc/logstash/conf.d/
0	2284	bash	2025-12-01 21:11:16.000000 UTC	priority: 1,
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo nano /etc/kibana/kibana.yml
0	2284	bash	2025-12-01 21:11:16.000000 UTC	settings: {
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo apt update
0	2284	bash	2025-12-01 21:11:16.000000 UTC	ls -l /etc/logstash/conf.d/
0	2284	bash	2025-12-01 21:11:16.000000 UTC	nc -zv <tu-ip> 514
0	2284	bash	2025-12-01 21:11:16.000000 UTC	nc -zv 192.168.100.71 514

0	2284	bash	2025-12-01 21:11:16.000000 UTC	cd ~
0	2284	bash	2025-12-01 21:11:16.000000 UTC	curl -L -O https://artifacts.elastic.co/
				downloads/beats/elastic-agent/
				elastic-agent-8.11.1-linux-x86_64.tar.gz
0	2284	bash	2025-12-01 21:11:16.000000 UTC	tar xzvf elastic-agent-8.11.1-linux-x86_64.tar.gz
0	2284	bash	2025-12-01 21:11:16.000000 UTC	cd elastic-agent-8.19.7-linux-x86_64
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo ./elastic-agent install
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo /opt/Elastic/Agent/elastic-agent uninstall
0	2284	bash	2025-12-01 21:11:16.000000 UTC	cd~

0	2284	bash	2025-12-01 21:11:16.000000 UTC	cd ~
0	2284	bash	2025-12-01 21:11:16.000000 UTC	curl -X GET "localhost:9200/_cat/indices?v"
0	2284	bash	2025-12-01 21:11:16.000000 UTC	curl -u elastic:elksatckG7 -X GET "localhost:9200/_cat/indices?v"
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo ls -l /var/lib/logstash
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo nano /etc/logstash/conf.d/
				30-elasticsearch-output.conf
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo ufw allow 1514/udp
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo ufw status verbose
0	2284	bash	2025-12-01 21:11:16.000000 UTC	init 6

0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo systemctl status logstash
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo journalctl --vacuum-size=500M
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo journalctl --vacuum-time=1s
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo truncate -s 0 /var/log/syslog
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo truncate -s 0 /var/log/kern.log
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo truncate -s 0 /var/log/auth.log
0	2284	bash	2025-12-01 21:11:16.000000 UTC	,
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo mkdir -p /usr/share/logstash/config
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo nano /usr/share/logstash/

				config/pipelines.yml
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo netstat -tuln | grep 1514
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo ufw allow 514/udp
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo rm -rf /var/log/journal/*
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo journalctl --vacuum-time=1s
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo journalctl --vacuum-size=500M
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo systemctl restart rsyslog
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo systemctl restart elasticsearch
0	2284	bash	2025-12-01 21:11:16.000000 UTC	sudo nano /etc/logstash/conf.d/
				02-beats-input.conf

Procesos con jerarquía (más útil para detectar malware oculto)

Ejecutamos el siguiente comando

```
python3 vol.py -f /home/kali/Documentos/memdump.lime linux.pstree
```

Procesos sospechosos identificados

Tras revisar la salida, solo hay 2 elementos realmente sospechosos:

Cadena sospechosa vía SSH → sudo → insmod

```
sshd (PID 1854)
 └── sshd (2229)
      └── sshd (2283)
           └── bash (2284)
                └── sudo (3514)
                     └── sudo (3515)
                          └── insmod (3516)
```

¿Por qué es sospechoso?

insmod carga un módulo en el kernel.

Un atacante que obtiene acceso suele usar insmod para cargar:

- un rootkit
- un keylogger de kernel

- un módulo persistente oculto

Todo ocurre después de una sesión SSH esto indica actividad manual.

Alta probabilidad de compromiso.

Java con procesos hijo extraños

Observaciones relevantes:

```
java (PID 986)
java (PID 990)
 └─ java (1612)
      └─ controller (1639)
```

Estos son Indicadores sospechosos:

Tres procesos Java sin contexto (no parece un servidor web clásico, ni Elastic, ni Jenkins).

El proceso hijo controller no es un proceso típico en instalaciones normales de Java, Java suele ser usado como loader para malware de botnet (Mirai-Java, Xor-DDoS, Kinsing, etc.).

Esto es sospechoso y requiere análisis

Otros procesos anómalos, pero no necesariamente maliciosos

```
psimon (PID 522 y 1423)
```

Nombre inusual.

Puede ser un proceso legítimo de monitoreo, pero también coincide con algunos binarios usados para system fingerprinting.

Un proceso Node.js (PID 992)

Normal solo si el servidor corre servicios node.

¿Hay indicios de malware o backdoor?

Sí existen indicios fuertes

Carga de un módulo (insmod) tras acceso SSH

Esto es típico de rootkits tipo Diamorphine, Reptile, Adore-NG, Suterusu ocultación de procesos keyloggers de kernel

Comportamiento extraño en procesos Java

Muy posiblemente:

```
un crypto-miner,
un backdoor,
un loader para malware residente.
```

Tabla 5

Comportamiento extraño en procesos Java

PYTHON3 VOL.PY -F

/HOME/KALI/DOCUMENTOS/MEMDUMP.LIME LINUX.PSTREE

VOLATILITY 3 FRAMEWORK 2.27.0				
PROGRESS: 100.00	Stacking attempts finished			
OFFSET (V)	PID	TID	PPID	COMM
0X8ADFC08AA8C0	1	1	0	systemd
*** 0X8ADFCAFF5180**	447	447	1	systemd-journal
*** 0X8ADFC4F08000**	474	474	1	multipathd
*** 0X8ADFC4D7A8C0**	507	507	1	systemd-udevd
*** 0X8ADFC4E5A8C0**	881	881	1	systemd-network
*** 0X8ADFC4E5D180**	903	903	1	systemd-resolve
*** 0X8ADFC74ED180**	904	904	1	systemd-timesyn
*** 0X8ADFC4CE8000**	906	906	1	VGAuthService
*** 0X8ADFC53D8000**	907	907	1	vmtoolsd
*** 0X8ADFC4AC28C0**	971	971	1	dbus-daemon
*** 0X8ADFCC4F5180**	976	976	1	polkitd
*** 0X8ADFC7700000**	981	981	1	systemd-logind
*** 0X8ADFC39528C0**	982	982	1	udisksd
*** 0X8ADFC8ACA8C0**	986	986	1	java
*** 0X8ADFC74F8000**	990	990	1	java
**** 0X8ADFC4850000**	1612	1612	990	java
***** 0X8ADFCA9B8000**	1639	1639	1612	controller
*** 0X8ADFCC03A8C0**	991	991	1	filebeat
*** 0X8ADFC8ACD180**	992	992	1	node

*** 0X8ADFC91CA8C0**	994	994	1	cron
*** 0X8ADFC91CD180**	996	996	1	rsyslogd
*** 0X8ADFC8BC28C0**	1007	1007	1	unattended-upgr
*** 0X8ADFC86DD180**	1018	1018	1	login
**** 0X8ADFCAC328C0**	1470	1470	1018	bash
*** 0X8ADFCD39A8C0**	1294	1294	1	ModemManager
*** 0X8ADFCD68D180**	1425	1425	1	systemd
**** 0X8ADFC483A8C0**	1426	1426	1425	(sd-pam)
*** 0X8ADFC63628C0**	1797	1797	1	fwupd
*** 0X8ADFC08C5180**	1806	1806	1	upowerd
*** 0X8ADFC637D180**	1854	1854	1	sshd
**** 0X8ADFC138A8C0**	2229	2229	1854	sshd
***** 0X8AE0009A0000**	2283	2283	2229	sshd
****** 0X8ADFCC395180**	2284	2284	2283	bash
******* 0X8ADFC35ED180**	3514	3514	2284	sudo
******** 0X8ADFCC2FD180**	3515	3515	3514	sudo
********* 0X8ADF1F815180**	3516	3516	3515	insmod
*** 0X8ADFCC1E0000**	2591	2591	1	packagekitd
0X8ADFC08AD180	2	2	0	kthreadd
*** 0X8ADFC08A8000**	3	3	2	pool_workqueue_
*** 0X8ADFC08BA8C0**	4	4	2	kworker/R-rcu_g
*** 0X8ADFC08BD180**	5	5	2	kworker/R-rcu_p
*** 0X8ADFC08B8000**	6	6	2	kworker/R-slub_
*** 0X8ADFC08C28C0**	7	7	2	kworker/R-netns
*** 0X8ADFC08CA8C0**	12	12	2	kworker/R-mm_pe
*** 0X8ADFC09D0000**	13	13	2	rcu_tasks_kthre

* 0X8ADFC09D28C0	14	14	2	rcu_tasks_rude_
* 0X8ADFC09D5180	15	15	2	rcu_tasks_trace
* 0X8ADFC09DA8C0	16	16	2	ksoftirqd/0
* 0X8ADFC09DD180	17	17	2	rcu_preempt
* 0X8ADFC09D8000	18	18	2	migration/0
* 0X8ADFC09ED180	19	19	2	idle_inject/0
* 0X8ADFC0DAA8C0	20	20	2	cpuhp/0
* 0X8ADFC0DB5180	21	21	2	cpuhp/1
* 0X8ADFC0DB0000	22	22	2	idle_inject/1
* 0X8ADFC0DB28C0	23	23	2	migration/1
* 0X8ADFC0DBA8C0	24	24	2	ksoftirqd/1
* 0X8ADFC0DCA8C0	27	27	2	cpuhp/2
* 0X8ADFC0DCD180	28	28	2	idle_inject/2
* 0X8ADFC0DC8000	29	29	2	migration/2
* 0X8ADFC0DD5180	30	30	2	ksoftirqd/2
* 0X8ADFC0DEA8C0	33	33	2	kdevtmpfs
* 0X8ADFC0E85180	34	34	2	kworker/R-inet_
* 0X8ADFC0E80000	35	35	2	kauditd
* 0X8ADFC1388000	39	39	2	khungtaskd
* 0X8ADFC19D28C0	41	41	2	oom_reaper
* 0X8ADFC19D0000	43	43	2	kworker/R-write
* 0X8ADFC19DD180	44	44	2	kcompactd0
* 0X8ADFC19D8000	45	45	2	ksmd
* 0X8ADFC19DA8C0	46	46	2	khugepaged
* 0X8ADFC19EA8C0	47	47	2	kworker/R-kinte
* 0X8ADFC19ED180	48	48	2	kworker/R-kbloc
* 0X8ADFC19E8000	49	49	2	kworker/R-blkcg

* 0X8ADFC1AC0000	50	50	2	irq/9-acpi
* 0X8ADFC2380000	51	51	2	kworker/R-tpm_d
* 0X8ADFC23828C0	52	52	2	kworker/R-ata_s
* 0X8ADFC2385180	53	53	2	kworker/R-md
* 0X8ADFC23928C0	54	54	2	kworker/R-md_bi
* 0X8ADFC2395180	55	55	2	kworker/R-edac-
* 0X8ADFC2390000	56	56	2	kworker/R-devfr
* 0X8ADFC239A8C0	57	57	2	watchdogd
* 0X8ADFC1AC28C0	59	59	2	kswapd0
* 0X8ADFC2398000	60	60	2	ecryptfs-kthrea
* 0X8ADFC1AC5180	61	61	2	kworker/R-kthro
* 0X8ADFC290A8C0	62	62	2	irq/24-pciehp
* 0X8ADFC253A8C0	63	63	2	irq/25-pciehp
* 0X8ADFC290D180	64	64	2	irq/26-pciehp
* 0X8ADFC253D180	65	65	2	irq/27-pciehp
* 0X8ADFC2908000	66	66	2	irq/28-pciehp
* 0X8ADFC2915180	67	67	2	irq/29-pciehp
* 0X8ADFC2538000	68	68	2	irq/30-pciehp
* 0X8ADFC2910000	69	69	2	irq/31-pciehp
* 0X8ADFC2548000	70	70	2	irq/32-pciehp
* 0X8ADFC29128C0	71	71	2	irq/33-pciehp
* 0X8ADFC254A8C0	72	72	2	irq/34-pciehp
* 0X8ADFC39F8000	73	73	2	irq/35-pciehp
* 0X8ADFC254D180	74	74	2	irq/36-pciehp
* 0X8ADFC39FA8C0	75	75	2	irq/37-pciehp
* 0X8ADFC4210000	76	76	2	irq/38-pciehp
* 0X8ADFC39FD180	77	77	2	irq/39-pciehp

*** 0X8ADFC42128C0**	78	78	2	irq/40-pciehp
*** 0X8ADFC3A08000**	79	79	2	irq/41-pciehp
*** 0X8ADFC4215180**	80	80	2	irq/42-pciehp
*** 0X8ADFC3A0A8C0**	81	81	2	irq/43-pciehp
*** 0X8ADFC4218000**	82	82	2	irq/44-pciehp
*** 0X8ADFC3A0D180**	83	83	2	irq/45-pciehp
*** 0X8ADFC421A8C0**	84	84	2	irq/46-pciehp
*** 0X8ADFC3A10000**	85	85	2	irq/47-pciehp
*** 0X8ADFC421D180**	86	86	2	irq/48-pciehp
*** 0X8ADFC3A128C0**	87	87	2	irq/49-pciehp
*** 0X8ADFC4225180**	88	88	2	irq/50-pciehp
*** 0X8ADFC3A15180**	89	89	2	irq/51-pciehp
*** 0X8ADFC4220000**	90	90	2	irq/52-pciehp
*** 0X8ADFC3A1D180**	91	91	2	irq/53-pciehp
*** 0X8ADFC42228C0**	92	92	2	irq/54-pciehp
*** 0X8ADFC3A18000**	93	93	2	irq/55-pciehp
*** 0X8ADFC422D180**	95	95	2	kworker/R-acpi_
*** 0X8ADFC4228000**	96	96	2	scsi_eh_0
*** 0X8ADFC632A8C0**	97	97	2	kworker/R-scsi_
*** 0X8ADFC632D180**	98	98	2	scsi_eh_1
*** 0X8ADFC6328000**	99	99	2	kworker/R-scsi_
*** 0X8ADFC6365180**	102	102	2	kworker/R-mld
*** 0X8ADFC6378000**	104	104	2	kworker/R-ipv6_
*** 0X8ADFC53DA8C0**	112	112	2	kworker/R-kstrp
*** 0X8ADFC53DD180**	114	114	2	kworker/u257:0
*** 0X8ADFC53E0000**	119	119	2	kworker/R-crypt
*** 0X8ADFCAFF0000**	130	130	2	kworker/R-charg

* **0X8ADFCB61D180**	181	181	2	kworker/2:1H
* **0X8ADFCB618000**	183	183	2	kworker/R-mpt_p
* **0X8ADFCB575180**	184	184	2	kworker/R-mpt/0
* **0X8ADFCB570000**	185	185	2	scsi_eh_2
* **0X8ADFCB5728C0**	186	186	2	kworker/R-scsi_
* **0X8ADFCB080000**	187	187	2	scsi_eh_3
* **0X8ADFCB0828C0**	188	188	2	kworker/R-scsi_
* **0X8ADFCB085180**	189	189	2	scsi_eh_4
* **0X8ADFCB37A8C0**	190	190	2	kworker/R-scsi_
* **0X8ADFCB37D180**	191	191	2	scsi_eh_5
* **0X8ADFCB378000**	192	192	2	kworker/R-scsi_
* **0X8ADFCB498000**	193	193	2	scsi_eh_6
* **0X8ADFCB49A8C0**	194	194	2	kworker/R-scsi_
* **0X8ADFCAEDD180**	195	195	2	scsi_eh_7
* **0X8ADFCC015180**	196	196	2	kworker/R-scsi_
* **0X8ADFCAED8000**	197	197	2	scsi_eh_8
* **0X8ADFCAEDA8C0**	198	198	2	kworker/R-scsi_
* **0X8ADFCAEEA8C0**	199	199	2	scsi_eh_9
* **0X8ADFCAEED180**	200	200	2	kworker/R-scsi_
* **0X8ADFCB3D0000**	201	201	2	scsi_eh_10
* **0X8ADFCAEE8000**	202	202	2	kworker/R-scsi_
* **0X8ADFCC085180**	210	210	2	scsi_eh_11
* **0X8ADFCC088000**	214	214	2	kworker/R-scsi_
* **0X8ADFCC08A8C0**	215	215	2	scsi_eh_12
* **0X8ADFCB3CA8C0**	216	216	2	kworker/R-scsi_
* **0X8ADFCB3CD180**	217	217	2	scsi_eh_13
* **0X8ADFCB3C5180**	219	219	2	kworker/R-scsi_

* 0X8ADFCB3C0000	221	221	2	scsi_eh_14
* 0X8ADFCAEF28C0	222	222	2	kworker/R-scsi_
* 0X8ADFCAEF0000	223	223	2	scsi_eh_15
* 0X8ADFCC1BD180	224	224	2	kworker/R-scsi_
* 0X8ADFCC1B8000	225	225	2	scsi_eh_16
* 0X8ADFCC1BA8C0	226	226	2	kworker/R-scsi_
* 0X8ADFCC1C8000	227	227	2	scsi_eh_17
* 0X8ADFCC0B5180	228	228	2	kworker/R-scsi_
* 0X8ADFCC0B0000	229	229	2	scsi_eh_18
* 0X8ADFCC27A8C0	234	234	2	kworker/R-scsi_
* 0X8ADFCC10A8C0	235	235	2	scsi_eh_19
* 0X8ADFCC10D180	236	236	2	kworker/R-scsi_
* 0X8ADFCC108000	237	237	2	scsi_eh_20
* 0X8ADFCC1CA8C0	238	238	2	kworker/R-scsi_
* 0X8ADFCC1CD180	239	239	2	scsi_eh_21
* 0X8ADFCC0B28C0	240	240	2	kworker/R-scsi_
* 0X8ADFCC08D180	241	241	2	scsi_eh_22
* 0X8ADFCC27D180	242	242	2	kworker/R-scsi_
* 0X8ADFCAEF5180	243	243	2	scsi_eh_23
* 0X8ADFCB3D28C0	244	244	2	kworker/R-scsi_
* 0X8ADFCC0828C0	245	245	2	scsi_eh_24
* 0X8ADFCC080000	246	246	2	kworker/R-scsi_
* 0X8ADFCB3C8000	247	247	2	scsi_eh_25
* 0X8ADFCB3C28C0	248	248	2	kworker/R-scsi_
* 0X8ADFCC155180	249	249	2	scsi_eh_26
* 0X8ADFCC150000	250	250	2	kworker/R-scsi_
* 0X8ADFCC1528C0	251	251	2	scsi_eh_27

* **0X8ADFCC2A8000**	252	252	2	kworker/R-scsi_
* **0X8ADFCC2AA8C0**	253	253	2	scsi_eh_28
* **0X8ADFCC2AD180**	254	254	2	kworker/R-scsi_
* **0X8ADFCC2C5180**	255	255	2	scsi_eh_29
* **0X8ADFCC2C0000**	256	256	2	kworker/R-scsi_
* **0X8ADFCC2C28C0**	257	257	2	scsi_eh_30
* **0X8ADFCC2D5180**	258	258	2	kworker/R-scsi_
* **0X8ADFCC2D0000**	259	259	2	scsi_eh_31
* **0X8ADFCC2D28C0**	260	260	2	kworker/R-scsi_
* **0X8ADFCC390000**	279	279	2	kworker/u256:24
* **0X8ADFCC1E5180**	287	287	2	scsi_eh_32
* **0X8ADFCC3AA8C0**	288	288	2	kworker/R-scsi_
* **0X8ADFCC020000**	311	311	2	kworker/R-kdmfl
* **0X8ADFCC038000**	340	340	2	kworker/R-raid5
* **0X8ADFCD398000**	379	379	2	jbd2/dm-0-8
* **0X8ADFCC03D180**	380	380	2	kworker/R-ext4-
* **0X8ADFCD68A8C0**	472	472	2	kworker/R-kmpat
* **0X8ADFCD688000**	473	473	2	kworker/R-kmpat
* **0X8ADFC4CEA8C0**	522	522	2	psimon
* **0X8ADFC395A8C0**	569	569	2	irq/57-vmw_vmci
* **0X8ADFC3958000**	570	570	2	irq/58-vmw_vmci
* **0X8ADFC3A1A8C0**	571	571	2	irq/59-vmw_vmci
* **0X8ADFC3950000**	576	576	2	irq/16-vmwgfx
* **0X8ADFC3955180**	577	577	2	kworker/R-ttm
* **0X8ADFC4AC5180**	691	691	2	jbd2/sda2-8
* **0X8ADFC4AC0000**	692	692	2	kworker/R-ext4-
* **0X8ADFC74E8000**	885	885	2	kworker/R-cfg80

* 0X8ADFC4838000	1423	1423	2	psimon
* 0X8ADFC0DE8000	1767	1767	2	kworker/1:1
* 0X8ADF60030000	1842	1842	2	kworker/u256:0
* 0X8AE0008028C0	1940	1940	2	kworker/1:0H
* 0X8AE0009A28C0	1948	1948	2	kworker/0:2
* 0X8AE0009C0000	1986	1986	2	kworker/u256:1
* 0X8ADFEAF30000	2079	2079	2	kworker/1:2
* 0X8ADF1F975180	2098	2098	2	kworker/1:2H
* 0X8ADFC37A28C0	2185	2185	2	kworker/2:0
* 0X8ADF60078000	2202	2202	2	kworker/2:0H
* 0X8ADED0028000	2215	2215	2	kworker/0:1H
* 0X8ADED00F5180	2220	2220	2	kworker/u256:4
* 0X8ADFCC1E28C0	2329	2329	2	kworker/0:0
* 0X8ADFC08C0000	2330	2330	2	kworker/2:1
* 0X8AE000E2D180	2626	2626	2	kworker/0:0H
* 0X8AE000800000	2640	2640	2	kworker/u256:2
* 0X8AE000805180	2641	2641	2	kworker/u256:3
* 0X8ADF1F8128C0	3518	3518	2	kworker/1:1H
* 0X8ADF1F810000	3519	3519	2	kworker/2:2H
* 0X8AE0009C28C0	3520	3520	2	kworker/u256:5
* 0X8ADFC35E8000	3522	3522	2	kworker/u256:6
* 0X8ADFC422A8C0	3524	3524	2	kworker/0:2H

Llaves de ejecución y binarios cargados (muy útil para rootkits)

Ejecutamos el siguiente comando

```
python3 vol.py -f /home/kali/Documentos/memdump.lime linux.lsof
```

Este comando nos da un volcado de descriptor de archivos abiertos (lsof o salida similar) de un sistema Linux el cual cntiene información sobre:

PID (Process ID)

PPID (Parent PID)

Nombre del proceso

FD (file descriptor)

Tipo de archivo (regular, socket, pipe, character device, etc.)

Estado de permisos

Tiempos de acceso, modificación y cambio

Tamaño del archivo

Esto nos permite ver:

Qué procesos están activos.

Qué recursos (archivos, sockets, pipes) están usando.

Posibles interacciones entre procesos (IPC, sockets, pipes).

Indicadores de actividad de usuario y del sistema.

Procesos destacados

gdbus (PID 1806, 1812, 2594, etc.)

Varias instancias.

FD de tipo anon_inode y socket.

Usa sockets para comunicación interprocesos (SOCK).

El tiempo de actividad parece estable (timestamps consistentes).

Indicador: proceso normal del sistema, probablemente parte de systemd o DBus.

sshd (PID 1854, 2229, 2283)

Múltiples instancias de SSH demonio, con redirección de STDIN/STDOUT a /dev/null o a pts, varias conexiones activas vía sockets (SOCK).

Se observa /dev/ptmx y /dev/pts/X, lo que indica sesiones de terminal activas.

Sesiones abiertas referenciadas por /run/systemd/sessions.

Esto indica usuarios conectados por SSH, probablemente el administrador o conexiones remotas activas.

packagekitd (PID 2591) y subprocesos pool-spawner / gmain
Múltiples FDs abiertos a sockets y anon_inodes.

Acceso a /var/lib/PackageKit/transactions.db a base de datos de transacciones de actualizaciones.

Proceso típico del sistema para manejar actualizaciones de paquetes.

FDs de tipo anon_inode sugieren uso de epoll o eventfd, típico de servicios que esperan eventos.

bash (PID 2284)

Terminal /dev/pts/0

STDIN, STDOUT, STDERR correctamente asociados

FD 255 también en /dev/pts/0, probablemente sesión interactiva de Bash.

sudo (PID 3514, 3515)

Ejecutando comandos con elevación de privilegios.

Acceso a /etc/sudoers y pipes de comunicación.

Se observa interacción con /dev/pts/0 y /dev/pts/1, esto es probablemente ejecución de insmod para cargar un módulo.

insmod (PID 3516)

Archivo cargado: /home/elk/LiME/src/lime.ko, tamaño: 1.95 MB

Esto indica carga de un módulo de kernel, en este caso **LiME (Linux Memory Extractor).**

Esto confirma que se está realizando un volcado de memoria del sistema con fines forenses o de análisis de incidentes.

Observaciones sobre seguridad / incidentes

Cualquier carga de módulo de kernel con sudo es sensible, LiME es un módulo de memoria legítimo, pero puede ser usado por un atacante si tiene acceso a sudo.

Las sesiones SSH activas sugieren que hay al menos una conexión remota activa, Sockets y pipes abiertos entre procesos sugieren comunicación normal de servicios (PackageKit, gdbus) y actividades del sistema.

Archivos de configuración sensibles (/etc/sudoers) están correctamente protegidos (-r--r-----).

Línea de tiempo de eventos importantes

20:26 UTC → inicio de muchos procesos del sistema (sshd, gdbus, packagekitd).

21:11–21:18 UTC → actividad de usuario en Bash y sudo.

21:17–21:18 UTC → carga de módulo LiME (insmod)

21:18 UTC → posibles pipes creados para comunicación con insmod / sudo.

Esto indica que la actividad del usuario y la extracción de memoria se realizó después del inicio de los servicios.

Resumen de patrones de actividad

Tabla 6

Resumen de patrones de actividad

PROCESO	ACTIVIDAD DESTACADA	SEGURIDAD / NOTAS
GDBUS	Comunicación IPC, anon_inodes, sockets	Normal
SSHD	Sesiones abiertas, terminales /dev/pts	Posible vector remoto si hay intrusos
PACKAGEKITD	Acceso a DB de transacciones, sockets	Normal
BASH	Terminal interactiva	Normal
SUDO	Elevación de privilegios	Riesgo si no es usuario autorizado
INSMOD	Carga de LiME (módulo kernel)	Acción forense legítima o riesgo si malicioso

Buscar módulos del kernel cargados (rootkits)

Se ejecuto el siguiente comando

```
python3 vol.py -f /home/kali/Documentos/memdump.lime linux.lsmod
```

Esto lista los módulos del kernel cargados en la memoria de un sistema Linux a partir de un volcado (memdump.lime).

Cada fila muestra información sobre un módulo del kernel, incluyendo su dirección en memoria (Offset), nombre (Module Name), tamaño de código (Code Size), indicadores de seguridad (Taints), argumentos de carga (Load Arguments) y salida de archivo (File Output).

Análisis general

Módulos del sistema y de red:

ip_tables, x_tables, nf_conntrack, xt_LOG, xt_limit son módulos de filtrado y logging de paquetes.

cfg80211, vsock, vmw_vsock_vmci_transport son módulos relacionados con red, virtualización y Wi-Fi.

Módulos de almacenamiento y RAID:

btrfs, raid0, raid1, raid10, raid456 son controladores de sistemas de archivos y arreglos RAID.

ahci, libahci, pata_acpi son controladores de almacenamiento SATA/IDE.

Módulos de hardware y sonido:

snd, snd_pcm, snd_timer, snd_seq_device son drivers de sonido.

intel_rapl_msr, intel_rapl_common son drives de monitoreo de energía de CPU Intel.

joydev, psmouse, usbhid, hid son periféricos de entrada (ratón, teclado, joysticks).

Criptografía y hashing:

aesni_intel, sha256_ssse3, sha1_ssse3, libcrc32c, aceleración de cifrado y hashing.

Virtualización (VMware):

vmwgfx, vmw_balloon, vmw_vmci son módulos típicos de máquinas virtuales VMware.

Módulos con advertencias de seguridad:

LiME es el módulo de Volatility que se carga para permitir el volcado de memoria, marcado como OUT OF TREE y UNSIGNED (no firmado por el kernel).

Esto puede indicar manipulación o análisis forense.

Tabla resumida de los módulos más importantes

Tabla 7

Tabla resumida de los módulos más importantes

CATEGORÍA	MÓDULO(S) CLAVE	NOTAS IMPORTANTES
VOLATILITY / ANÁLISIS	lime	OUT OF TREE, UNSIGNED, usado para volcado de memoria
RED Y RED VIRTUAL	cfg80211, vsock, vmw_vsock_vmci_transport	Controladores de red, virtualización y Wi-Fi
FILTRADO DE PAQUETES / FIREWALL	ip_tables, x_tables, nf_conntrack, xt_LOG	Filtrado de tráfico y logging
ALMACENAMIENTO / RAID	btrfs, raid0, raid1, raid10, raid456, ahci, libahci	Sistemas de archivos, controladores SATA/IDE
SONIDO Y MULTIMEDIA	snd, snd_pcm, snd_timer, snd_seq_device	Drivers de audio
PERIFÉRICOS DE ENTRADA	psmouse, usbhid, hid, joydev	Teclado, ratón, joysticks
CRIPTOGRAFÍA / HASHING	aesni_intel, sha256_ssse3, sha1_ssse3, libcrc32c	Aceleración de cifrado y hashing
VIRTUALIZACIÓN VMWARE	vmwgfx, vmw_balloon, vmw_vmci	Controladores de VM
ENERGÍA / CPU	intel_rapl_msr, intel_rapl_common	Monitoreo de energía de CPU Intel

Buscar inyecciones de código / memoria anómala

Ejecutamos el siguiente comando

```
python3 vol.py -f /home/kali/Documentos/memdump.lime linux.malfind
```

Análisis general

Plugin usado: linux.malfind

Detecta regiones de memoria anónimas (no respaldadas por archivos) con permisos ejecutables (rwx o r-x).

Estas regiones suelen asociarse a inyecciones de código, shellcode o malware en memoria.

Procesos afectados:

PID 986 (java): múltiples regiones sospechosas.

PID 990 (java): varias regiones de memoria anónima rwx.

PID 992 (node): memoria anónima rwx también detectada.

Características de las regiones:

Muchas regiones tienen permisos rwx esto indica que se pueden leer, escribir y ejecutar (comportamiento típico de shellcode/memoria inyectada).

Se observan hexdumps repetitivos con operaciones como add byte ptr [rax], esto indica patrón de NOP sled o código polimórfico.

Algunas regiones tienen r-x, lo que indica solo lectura y ejecución, común en código legítimo cargado dinámicamente.

Patrones sospechosos:

Regiones grandes y anónimas:

0x714f942c7000 - 0x714f959e7000 (986 java) tiene aproximadamente 1.7 MB, permisos rwx

0x78273c0c6000 - 0x78273c336000 (990 java) tiene aproximadamente 3.1 MB, permisos rwx

Estos tamaños y permisos son inusuales para objetos de Java y Node normales, lo que refuerza la hipótesis de código inyectado.

Los procesos java y node son comúnmente atacados por malware debido a que corren aplicaciones de alto nivel con acceso a la memoria y a la red.

Disassembly:

Muchas instrucciones repetitivas (add byte ptr [rax], al) esto es típico de código ofuscado o shellcode polimórfico para evitar detección por antivirus/memoria.

Algunas instrucciones (call, mov, jne) sugieren funcionalidad real inyectada, pero el patrón repetitivo indica manipulación de memoria o NOP sled.

Tabla 8

Resultados del desensmblado

PID	PROCESO	DIRECCIÓN INICIO	DIRECCIÓN FIN	TAMAÑO APROXIMADO	PERMISOS	OBSERVACIONES CLAVE
986	java	0x714f8c800000	0x714f90290000	1.7 MB	rwx	Región anónima ejecutable, posibles shellcode, instrucciones repetitivas y ofuscadas.
986	java	0x714f93d38000	0x714f93fa8000	~1.7 MB	rwx	Otra región anónima con patrón NOP sled, código inyectado probable.
986	java	0x714f942c7000	0x714f959e7000	~1.7 MB	rwx	Código ofuscado, instrucciones repetitivas, inyección de memoria sospechosa.
986	java	0x714facbbc000	0x714facbbf000	~60 KB	r-x	Posible código legítimo, pero mezcla de NOP sled y llamadas internas sospechosas.
990	java	0x782734600000	0x782734a80000	~288 KB	rwx	Memoria anónima, patrón de NOP sled, inyección potencial.
990	java	0x78273bb38000	0x78273bda8000	~3.2 MB	rwx	Muy grande, permisos completos, código posiblemente malicioso.
990	java	0x78273c0c6000	0x78273c336000	~3.1 MB	rwx	Misma observación que anterior, alta probabilidad de

						malware en memoria.
992	node	0xaf60b85b000	0xaf60b85c000	4 KB	rwx	Región pequeña, inyección potencial.
992	node	0x214b795dd000	0x214b795de000	4 KB	rwx	Región pequeña, código inyectado en memoria de Node.js.

Relleno repetitivo con add byte ptr [rax],

Este patrón aparece en casi todas las regiones (0x7f55e4183000, 0x7f55e41c3000, 0x7f55e4203000, etc.).

Este tipo de secuencia suele ser indicativa de padding o shellcode ofuscado. En malware o exploits, se utiliza para alinear el código o confundir el análisis estático.

También podría ser simplemente espacio reservado en memoria (una especie de “NOP sled” usando add en lugar de nop).

Acceso a memoria con registros no inicializados (rax, rcx, rsi)

Muchas instrucciones operan sobre direcciones como [rax] o [rcx + offset] sin que se vea un valor seguro cargado antes.

Esto es típico de código auto-modificante o shellcode dinámico que manipula su propia memoria.

Uso de instrucciones extrañas y poco comunes

int 0x38, fnop, fcmovb st(0), st(1), in al, dx

Estas instrucciones pueden ser anti-debug o anti-análisis.

Memoria RWX

Todas las regiones mapeadas (Anonymous Mapping rwx) permiten lectura, escritura y ejecución, lo cual es un signo típico de ejecución dinámica de código, muy común en exploits, cargadores de malware o JIT shells.

Patrones de “mov + sbb + add” con valores constantes

Secuencias como:

mov ecx, 0xe445a411

sbb al, byte ptr [rax]

add byte ptr [rcx + offset], reg

Todo esto apunta a manipulación de memoria calculada.

Posiblemente estaban decodificando datos cifrados o construyendo instrucciones en tiempo de ejecución.

Tabla 9

Detección de patrones en tiempo de ejecución

# PARTE	DIRECCIÓN / OFFSET	PATRÓN PRINCIPAL	OBSERVACIONES CLAVE	INDICADOR DE MALWARE
1	0x7f55e4a03000	add [rax], al	Relleno / NOP sled	Alto
2	0x7f55e4a43000	add [rax], al + mov/sbb	Inicio preparación payload	Alto
3/4	0x7f55e4a83000 / 0x7f55e4ac3000	add [rax], al + sbb/adc/rcr/xor/cwde	Stub de decodificación, núcleo crítico, anti-debug	Muy alto
5	0x7f55e4b03000	add [rax], al + push/lea/jmp	Preparación final del payload	Muy alto
6	0x7f55e4b43000	add [rax], al + xor/jmp	Anti-disassembly, manipulación de registros	Muy alto
7	0x7f55e4b83000	add [rax], al + adc / push / add	Relleno + preparación de ejecución	Alto
8-10	0x758466001000	add [rax], al	Payload final en memoria, RWX, page “malicious”	Muy alto

Análisis Forense de la RAM memdump.lime con Volatility3

Datos Generales de la Evidencia

Nombre del archivo analizado: memdump.lime

Hash MD5: 87e4be4c6c7a808af6212e4a69faa57f

Hash SHA256:

eb90256449b9fd738fc76c5c437fe897e8dcdeedc20f70fd46d4281f13c089d2

Fecha de adquisición: 02/12/2025

Herramienta utilizada: Volatility WorkBench

Analista responsable: grupo7

Como podemos observar el hash MD5 al ser procesado en el repositorio para su análisis con fecha 02-12-2025 es el mismo que se obtuvo junto con la imagen extraída, esto nos indica que la línea de tiempo, custodia y procesamiento no han sido corrompidas

Figura 172

Hash MD5 resultante del proceso con Volatility WorkBench

```
PS G:\PARA-FORENCE-PBL> CertUtil -hashfile '.\memdump.lime' MD5
MD5 hash de .\memdump.lime:
87e4be4c6c7a808af6212e4a69faa57f
CertUtil: -hashfile comando completado correctamente.
PS G:\PARA-FORENCE-PBL> CertUtil -hashfile '.\memdump.lime' SHA256
SHA256 hash de .\memdump.lime:
eb90256449b9fd738fc76c5c437fe897e8dcdeedc20f70fd46d4281f13c089d2
CertUtil: -hashfile comando completado correctamente.
```

6. Conclusiones y Recomendaciones

El análisis forense realizado sobre la imagen elk_var_particion.dd correspondiente al servidor SIEM basado en ELK permitió identificar información relevante sobre la actividad de usuarios, la operación del sistema y la posible presencia de eventos relacionados con ataques intencionales dentro del entorno de laboratorio.

Inicialmente, se revisaron numerosos archivos de configuración del sistema (incluyendo scripts de instalación de sudo, shadow, passwd, Filebeat y Kibana). Estos archivos no contenían información sobre usuarios conectados, sino configuraciones internas del sistema y servicios vinculados al SIEM. Por ello, se determinó que la evidencia de accesos debía buscarse específicamente dentro de los archivos de log del sistema.

Para ello, se configuró Autopsy para realizar búsquedas mediante la herramienta Keyword Search, la cual permite indexar e identificar cadenas específicas dentro de los logs. Se diseñó una lista de palabras clave orientadas tanto a la autenticación de usuarios como a la detección de ataques, incluyendo términos como Accepted, Failed, session opened, hydra, nmap, sshd, password, pam_unix y otros patrones relevantes. Esta lista se aplicó sobre los directorios de logs más importantes del sistema: /var/log/auth.log, /var/log/syslog, /var/log/kibana/, /var/log/elasticsearch/, entre otros.

Durante el análisis del contenido del disco, se identificaron varios fragmentos de logs relevantes que indicaban:

Sesiones de usuario legítimas asociadas al usuario elk, quien parecía ser el usuario principal administrador del servidor.

Registros de autenticación exitosos y fallidos relacionados con servicios como sshd, systemd, y pam_unix.

Evidencia explícita del nombre del atacante de laboratorio (“atacanteg7”), lo cual confirma que el entorno contenía simulaciones controladas de intrusiones.

Actividades registradas por Filebeat y Logstash, compatibles con un entorno SIEM en funcionamiento.

Presencia de patrones textuales que sugieren intentos de autenticación —incluyendo fallos y aciertos—, que coinciden con ataques automatizados como los realizados mediante Hydra, así como posibles escaneos o reconocimiento inicial típicos de herramientas como Nmap y Msfconsole.

Aunque no se detectaron archivos explícitos con comandos o payloads propios de exploits, los fragmentos encontrados demuestran de forma clara que la imagen contiene registros de intentos de intrusión, accesos fallidos, accesos exitosos y actividad sospechosa, producto de las pruebas de ataque realizadas en el laboratorio.

Referencias Bibliográficas

BitLyft. (2023, May 4). *Is Elastic Stack (ELK) the best SIEM tool?* https://www.bitlyft.com/resources/is-elastic-stack-elk-the-best-siem-tool

ChaosSearch. (2021, June 30). *ELK Stack SIEM: Everything you need to know*. https://www.chaossearch.io/blog/elk-stack-siem

Hiberus. (2022, September 14). *Qué es un stack de ELK y qué ventajas tiene implementarlo*. https://www.hiberus.com/crecemos-contigo/que-es-un-stack-de-elk-y-que-ventajas-tiene-implementarlo/

Organización de los Estados Americanos. (s. f.). *Análisis forense digital: Manual práctico*. https://www.oas.org/juridico/spanish/cyb_analisis_foren.pdf

Universidad Piloto de Colombia. (2015). *Propuesta de implementación de una solución SIEM basada en ELK Stack* [Tesis de pregrado]. http://polux.unipiloto.edu.co:8080/00004741.pdf

Greenwebpage. (2024). *How to enable SSH on Ubuntu 24.04*. https://greenwebpage.com/community/how-to-enable-ssh-on-ubuntu-24-04/

Elastic. (2024). *Elastic Stack documentation*. https://www.elastic.co/guide/index.html

Elastic. (2024). *Security features in Elasticsearch*. https://www.elastic.co/guide/en/elasticsearch/reference/current/security-settings.html

Elastic. (2024). *Audit logging*. https://www.elastic.co/guide/en/elasticsearch/reference/current/auditing-settings.html

Elastic. (2024). *Kibana security*. https://www.elastic.co/guide/en/kibana/current/xpack-security.html

Behl, A., & Behl, K. (2017). *Cyberwar: The next threat to national security and what to do about it*. Oxford University Press.

Scarfone, K., & Mell, P. (2007). *Guide to intrusion detection and prevention systems (SP 800-94)*. NIST. https://csrc.nist.gov

Chuvakin, A., Schmidt, K., & Phillips, C. (2013). *Logging and log management*. Syngress.

Splunk Inc. (2022). *SIEM fundamentals*. https://www.splunk.com

IBM. (2021). *Security information and event management (SIEM)*. https://www.ibm.com/security/siem

Carrier, B. (2005). *File system forensic analysis*. Addison-Wesley Professional.

Carrier, B., & Basis Technology. (2023). *Autopsy: The digital forensics platform*. https://www.autopsy.com

National Institute of Standards and Technology. (2006). *Computer security incident handling guide (SP 800-61 Rev. 2)*. https://doi.org/10.6028/NIST.SP.800-61r2

National Institute of Standards and Technology. (2014). *Guide to integrating forensic techniques into incident response (SP 800-86)*. https://doi.org/10.6028/NIST.SP.800-86

National Institute of Standards and Technology. (2012). *Guide to computer security log management (SP 800-92)*. https://csrc.nist.gov

National Institute of Standards and Technology. (2018). *Digital forensic science framework (NISTIR 8354)*. https://www.nist.gov

Prosise, C., & Mandia, K. (2003). *Incident response and computer forensics*. McGraw-Hill.

SANS Institute. (2020). *Digital forensics and incident response poster*. https://www.sans.org

Lyon, G. F. (2023). *Nmap network scanning*. Insecure.Com LLC. https://nmap.org/book/

Offensive Security. (2024). *Kali Linux documentation*. https://www.kali.org/docs/

Van Hauser, T. (2024). *THC Hydra*. https://github.com/vanhauser-thc/thc-hydra

Rapid7. (2024). *Metasploit Framework documentation*. https://docs.metasploit.com/

Stenberg, D. (2024). *curl documentation*. https://curl.se/docs/

Sylve, J., Case, A., Marziale, L., & Richard, G. G. (2012). Acquisition and analysis of volatile memory from Android devices. *Digital Investigation, 8*(3–4), 175–184.

504ensics Labs. (2024). *LiME – Linux Memory Extractor*. https://github.com/504ensicsLabs/LiME

Farmer, D., & Venema, W. (2005). *Forensic discovery*. Addison-Wesley.

SourceForge. (2024). *dcfldd*. https://dcfldd.sourceforge.net/

Kali Linux Tools. (2024). *Nikto*. https://www.kali.org/tools/nikto/

Kali Linux Tools. (2024). *Dirb*. https://www.kali.org/tools/dirb/

Lyon, G. F. (2023). *Nmap network scanning*. Insecure.Com LLC. https://nmap.org/book/

Nmap Project. (2024). *Nmap reference guide*. https://nmap.org/docs.html

Rapid7. (2024). *Metasploit Framework documentation*. https://docs.metasploit.com/

Sullo, C. (2024). *Nikto web scanner*. https://cirt.net/Nikto2

Kali Linux Tools. (2024). *Gobuster*. https://www.kali.org/tools/gobuster/

OWASP Foundation. (2021). *OWASP Top 10 Web Application Security Risks*. https://owasp.org

ANEXOS

Tabla 10

Indicadores de compromiso severo

YTHON3 VOL.PY -F /HOME/KALI/DOCUMENTOS/MEMDUMP.LIME LINUX.PSLIST

VOLATILITY 3 FRAMEWORK 2.27.0										
PROGRESS: 100.00	Stacking attempts finished									
OFFSET (V)	PID	TID	PPID	COMM	UID	GID	EUID	EGID	CREATION TIME	File output
0X8ADFC08AA8C0	1	1	0	systemd	0	0	0	0	2025-12-01 20:26:30.755237 UTC	Disabled
0X8ADFC08AD180	2	2	0	kthreadd	0	0	0	0	2025-12-01 20:26:30.755237 UTC	Disabled
0X8ADFC08A8000	3	3	2	pool_workqueue_	0	0	0	0	2025-12-01 20:26:30.755237 UTC	Disabled

0X8ADFC08 BA8C0	4	4	2	kworker/R-rcu_g	0	0	0	0	2025-12-01 20:26:30.755237 UTC	Disabled
0X8ADFC08 BD180	5	5	2	kworker/R-rcu_p	0	0	0	0	2025-12-01 20:26:30.755237 UTC	Disabled
0X8ADFC08 B8000	6	6	2	kworker/R-slub_	0	0	0	0	2025-12-01 20:26:30.755237 UTC	Disabled
0X8ADFC08 C28C0	7	7	2	kworker/R-netns	0	0	0	0	2025-12-01 20:26:30.755237 UTC	Disabled
0X8ADFC08 CA8C0	12	12	2	kworker/R-mm_pe	0	0	0	0	2025-12-01 20:26:30.756237 UTC	Disabled
0X8ADFC09 D0000	13	13	2	rcu_tasks_kthre	0	0	0	0	2025-12-01	Disabled

									20:26:30.756237 UTC	
0X8ADFC09D28C0	14	14	2	rcu_tasks_rude_	0	0	0	0	2025-12-01 20:26:30.756237 UTC	Disabled
0X8ADFC09D5180	15	15	2	rcu_tasks_trace	0	0	0	0	2025-12-01 20:26:30.757237 UTC	Disabled
0X8ADFC09DA8C0	16	16	2	ksoftirqd/0	0	0	0	0	2025-12-01 20:26:30.759237 UTC	Disabled
0X8ADFC09DD180	17	17	2	rcu_preempt	0	0	0	0	2025-12-01 20:26:30.759237 UTC	Disabled
0X8ADFC09D8000	18	18	2	migration/0	0	0	0	0	2025-12-01 20:26:30.7	Disabled

									59237 UTC	
0X8ADFC09 ED180	19	19	2	idle_inject/0	0	0	0	0	2025-12-01 20:26:30.760237 UTC	Disabled
0X8ADFC0D AA8C0	20	20	2	cpuhp/0	0	0	0	0	2025-12-01 20:26:30.775237 UTC	Disabled
0X8ADFC0D B5180	21	21	2	cpuhp/1	0	0	0	0	2025-12-01 20:26:30.775237 UTC	Disabled
0X8ADFC0D B0000	22	22	2	idle_inject/1	0	0	0	0	2025-12-01 20:26:30.775237 UTC	Disabled
0X8ADFC0D B28C0	23	23	2	migration/1	0	0	0	0	2025-12-01 20:26:30.776237 UTC	Disabled

0X8ADFC0DBA8C0	24	24	2	ksoftirqd/1	0	0	0	0	2025-12-01 20:26:30.776237 UTC	Disabled
0X8ADFC0DCA8C0	27	27	2	cpuhp/2	0	0	0	0	2025-12-01 20:26:30.777237 UTC	Disabled
0X8ADFC0DCD180	28	28	2	idle_inject/2	0	0	0	0	2025-12-01 20:26:30.777237 UTC	Disabled
0X8ADFC0DC8000	29	29	2	migration/2	0	0	0	0	2025-12-01 20:26:30.777237 UTC	Disabled
0X8ADFC0DD5180	30	30	2	ksoftirqd/2	0	0	0	0	2025-12-01 20:26:30.777237 UTC	Disabled
0X8ADFC0DEA8C0	33	33	2	kdevtmpfs	0	0	0	0	2025-12-01	Disabled

									20:26:30.780237 UTC	
0X8ADFC0E85180	34	34	2	kworker/R-inet_	0	0	0	0	2025-12-01 20:26:30.784237 UTC	Disabled
0X8ADFC0E80000	35	35	2	kauditd	0	0	0	0	2025-12-01 20:26:30.805237 UTC	Disabled
0X8ADFC1388000	39	39	2	khungtaskd	0	0	0	0	2025-12-01 20:26:30.809237 UTC	Disabled
0X8ADFC19D28C0	41	41	2	oom_reaper	0	0	0	0	2025-12-01 20:26:30.809237 UTC	Disabled
0X8ADFC19D0000	43	43	2	kworker/R-write	0	0	0	0	2025-12-01 20:26:30.8	Disabled

									09237 UTC	
0X8ADFC19DD180	44	44	2	kcompactd0	0	0	0	0	2025-12-01 20:26:30.813237 UTC	Disabled
0X8ADFC19D8000	45	45	2	ksmd	0	0	0	0	2025-12-01 20:26:30.813237 UTC	Disabled
0X8ADFC19DA8C0	46	46	2	khugepaged	0	0	0	0	2025-12-01 20:26:30.814237 UTC	Disabled
0X8ADFC19EA8C0	47	47	2	kworker/R-kinte	0	0	0	0	2025-12-01 20:26:30.814237 UTC	Disabled
0X8ADFC19ED180	48	48	2	kworker/R-kbloc	0	0	0	0	2025-12-01 20:26:30.815237 UTC	Disabled

0X8ADFC19E8000	49	49	2	kworker/R-blkcg	0	0	0	0	2025-12-01 20:26:30.815237 UTC	Disabled
0X8ADFC1AC0000	50	50	2	irq/9-acpi	0	0	0	0	2025-12-01 20:26:30.833237 UTC	Disabled
0X8ADFC2380000	51	51	2	kworker/R-tpm_d	0	0	0	0	2025-12-01 20:26:31.098237 UTC	Disabled
0X8ADFC23828C0	52	52	2	kworker/R-ata_s	0	0	0	0	2025-12-01 20:26:31.099237 UTC	Disabled
0X8ADFC2385180	53	53	2	kworker/R-md	0	0	0	0	2025-12-01 20:26:31.100237 UTC	Disabled
0X8ADFC23928C0	54	54	2	kworker/R-md_bi	0	0	0	0	2025-12-01	Disabled

									20:26:31.100237 UTC	
0X8ADFC2395180	55	55	2	kworker/R-edac-	0	0	0	0	2025-12-01 20:26:31.102237 UTC	Disabled
0X8ADFC2390000	56	56	2	kworker/R-devfr	0	0	0	0	2025-12-01 20:26:31.102237 UTC	Disabled
0X8ADFC239A8C0	57	57	2	watchdogd	0	0	0	0	2025-12-01 20:26:31.124237 UTC	Disabled
0X8ADFC1AC28C0	59	59	2	kswapd0	0	0	0	0	2025-12-01 20:26:31.244641 UTC	Disabled
0X8ADFC2398000	60	60	2	ecryptfs-kthrea	0	0	0	0	2025-12-01 20:26:31.2	Disabled

									45784 UTC	
0X8ADFC1AC5180	61	61	2	kworker/R-kthro	0	0	0	0	2025-12-01 20:26:31.257747 UTC	Disabled
0X8ADFC290A8C0	62	62	2	irq/24-pciehp	0	0	0	0	2025-12-01 20:26:31.259702 UTC	Disabled
0X8ADFC253A8C0	63	63	2	irq/25-pciehp	0	0	0	0	2025-12-01 20:26:31.261696 UTC	Disabled
0X8ADFC290D180	64	64	2	irq/26-pciehp	0	0	0	0	2025-12-01 20:26:31.263064 UTC	Disabled
0X8ADFC253D180	65	65	2	irq/27-pciehp	0	0	0	0	2025-12-01 20:26:31.264419 UTC	Disabled

0X8ADFC29 08000	66	66	2	irq/28-pciehp	0	0	0	0	2025-12-01 20:26:31.265996 UTC	Disabled
0X8ADFC29 15180	67	67	2	irq/29-pciehp	0	0	0	0	2025-12-01 20:26:31.267387 UTC	Disabled
0X8ADFC25 38000	68	68	2	irq/30-pciehp	0	0	0	0	2025-12-01 20:26:31.268711 UTC	Disabled
0X8ADFC29 10000	69	69	2	irq/31-pciehp	0	0	0	0	2025-12-01 20:26:31.270105 UTC	Disabled
0X8ADFC25 48000	70	70	2	irq/32-pciehp	0	0	0	0	2025-12-01 20:26:31.271486 UTC	Disabled
0X8ADFC29 128C0	71	71	2	irq/33-pciehp	0	0	0	0	2025-12-01	Disabled

									20:26:31.272904 UTC	
0X8ADFC254A8C0	72	72	2	irq/34-pciehp	0	0	0	0	2025-12-01 20:26:31.274219 UTC	Disabled
0X8ADFC39F8000	73	73	2	irq/35-pciehp	0	0	0	0	2025-12-01 20:26:31.275806 UTC	Disabled
0X8ADFC254D180	74	74	2	irq/36-pciehp	0	0	0	0	2025-12-01 20:26:31.277215 UTC	Disabled
0X8ADFC39FA8C0	75	75	2	irq/37-pciehp	0	0	0	0	2025-12-01 20:26:31.278572 UTC	Disabled
0X8ADFC4210000	76	76	2	irq/38-pciehp	0	0	0	0	2025-12-01 20:26:31.2	Disabled

									80182 UTC	
0X8ADFC39FD180	77	77	2	irq/39-pciehp	0	0	0	0	2025-12-01 20:26:31.281633 UTC	Disabled
0X8ADFC42128C0	78	78	2	irq/40-pciehp	0	0	0	0	2025-12-01 20:26:31.283020 UTC	Disabled
0X8ADFC3A08000	79	79	2	irq/41-pciehp	0	0	0	0	2025-12-01 20:26:31.284567 UTC	Disabled
0X8ADFC4215180	80	80	2	irq/42-pciehp	0	0	0	0	2025-12-01 20:26:31.285904 UTC	Disabled
0X8ADFC3A0A8C0	81	81	2	irq/43-pciehp	0	0	0	0	2025-12-01 20:26:31.287283 UTC	Disabled

0X8ADFC42 18000	82	82	2	irq/44-pciehp	0	0	0	0	2025-12-01 20:26:31.288695 UTC	Disabled
0X8ADFC3A 0D180	83	83	2	irq/45-pciehp	0	0	0	0	2025-12-01 20:26:31.290122 UTC	Disabled
0X8ADFC42 1A8C0	84	84	2	irq/46-pciehp	0	0	0	0	2025-12-01 20:26:31.291467 UTC	Disabled
0X8ADFC3A 10000	85	85	2	irq/47-pciehp	0	0	0	0	2025-12-01 20:26:31.293001 UTC	Disabled
0X8ADFC42 1D180	86	86	2	irq/48-pciehp	0	0	0	0	2025-12-01 20:26:31.294549 UTC	Disabled
0X8ADFC3A 128C0	87	87	2	irq/49-pciehp	0	0	0	0	2025-12-01	Disabled

									20:26:31.296004 UTC	
0X8ADFC4225180	88	88	2	irq/50-pciehp	0	0	0	0	2025-12-01 20:26:31.297493 UTC	Disabled
0X8ADFC3A15180	89	89	2	irq/51-pciehp	0	0	0	0	2025-12-01 20:26:31.298877 UTC	Disabled
0X8ADFC4220000	90	90	2	irq/52-pciehp	0	0	0	0	2025-12-01 20:26:31.300177 UTC	Disabled
0X8ADFC3A1D180	91	91	2	irq/53-pciehp	0	0	0	0	2025-12-01 20:26:31.301575 UTC	Disabled
0X8ADFC4228C0	92	92	2	irq/54-pciehp	0	0	0	0	2025-12-01 20:26:31.3	Disabled

									02867 UTC	
0X8ADFC3A 18000	93	93	2	irq/55-pciehp	0	0	0	0	2025-12-01 20:26:31.304114 UTC	Disabled
0X8ADFC42 2D180	95	95	2	kworker/R-acpi_	0	0	0	0	2025-12-01 20:26:31.307263 UTC	Disabled
0X8ADFC42 28000	96	96	2	scsi_eh_0	0	0	0	0	2025-12-01 20:26:31.410634 UTC	Disabled
0X8ADFC63 2A8C0	97	97	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:31.410951 UTC	Disabled
0X8ADFC63 2D180	98	98	2	scsi_eh_1	0	0	0	0	2025-12-01 20:26:31.411317 UTC	Disabled

0X8ADFC6328000	99	99	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:31.411451 UTC	Disabled
0X8ADFC6365180	102	102	2	kworker/R-mld	0	0	0	0	2025-12-01 20:26:31.433719 UTC	Disabled
0X8ADFC6378000	104	104	2	kworker/R-ipv6_	0	0	0	0	2025-12-01 20:26:31.434364 UTC	Disabled
0X8ADFC53DA8C0	112	112	2	kworker/R-kstrp	0	0	0	0	2025-12-01 20:26:31.760680 UTC	Disabled
0X8ADFC53DD180	114	114	2	kworker/u257:0	0	0	0	0	2025-12-01 20:26:31.786318 UTC	Disabled
0X8ADFC53E0000	119	119	2	kworker/R-crypt	0	0	0	0	2025-12-01	Disabled

									20:26:31.795039 UTC	
0X8ADFCAFF0000	130	130	2	kworker/R-charg	0	0	0	0	2025-12-01 20:26:31.819125 UTC	Disabled
0X8ADFCB61D180	181	181	2	kworker/2:1H	0	0	0	0	2025-12-01 20:26:32.068445 UTC	Disabled
0X8ADFCB618000	183	183	2	kworker/R-mpt_p	0	0	0	0	2025-12-01 20:26:32.307824 UTC	Disabled
0X8ADFCB575180	184	184	2	kworker/R-mpt/0	0	0	0	0	2025-12-01 20:26:32.309623 UTC	Disabled
0X8ADFCB570000	185	185	2	scsi_eh_2	0	0	0	0	2025-12-01 20:26:32.3	Disabled

									22595 UTC	
0X8ADFCB5728C0	186	186	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:32.323988 UTC	Disabled
0X8ADFCB080000	187	187	2	scsi_eh_3	0	0	0	0	2025-12-01 20:26:32.324501 UTC	Disabled
0X8ADFCB0828C0	188	188	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:32.324673 UTC	Disabled
0X8ADFCB085180	189	189	2	scsi_eh_4	0	0	0	0	2025-12-01 20:26:32.325202 UTC	Disabled
0X8ADFCB37A8C0	190	190	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:32.325451 UTC	Disabled

0X8ADFCB37D180	191	191	2	scsi_eh_5	0	0	0	0	2025-12-01 20:26:32.325990 UTC	Disabled
0X8ADFCB378000	192	192	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:32.326333 UTC	Disabled
0X8ADFCB498000	193	193	2	scsi_eh_6	0	0	0	0	2025-12-01 20:26:32.327159 UTC	Disabled
0X8ADFCB49A8C0	194	194	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:32.327182 UTC	Disabled
0X8ADFCAEDD180	195	195	2	scsi_eh_7	0	0	0	0	2025-12-01 20:26:32.327307 UTC	Disabled
0X8ADFCC015180	196	196	2	kworker/R-scsi_	0	0	0	0	2025-12-01	Disabled

									20:26:32.327882 UTC	
0X8ADFCAED8000	197	197	2	scsi_eh_8	0	0	0	0	2025-12-01 20:26:32.328318 UTC	Disabled
0X8ADFCADA8C0	198	198	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:32.328346 UTC	Disabled
0X8ADFCAEEA8C0	199	199	2	scsi_eh_9	0	0	0	0	2025-12-01 20:26:32.328477 UTC	Disabled
0X8ADFCAEED180	200	200	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:32.329475 UTC	Disabled
0X8ADFCB3D0000	201	201	2	scsi_eh_10	0	0	0	0	2025-12-01 20:26:32.3	Disabled

									30030 UTC	
0X8ADFCAEE8000	202	202	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:32.330346 UTC	Disabled
0X8ADFCC085180	210	210	2	scsi_eh_11	0	0	0	0	2025-12-01 20:26:32.337205 UTC	Disabled
0X8ADFCC088000	214	214	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:32.338945 UTC	Disabled
0X8ADFCC08A8C0	215	215	2	scsi_eh_12	0	0	0	0	2025-12-01 20:26:32.339318 UTC	Disabled
0X8ADFCB3CA8C0	216	216	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:32.339619 UTC	Disabled

0X8ADFCB3CD180	217	217	2	scsi_eh_13	0	0	0	0	2025-12-01 20:26:32.339880 UTC	Disabled
0X8ADFCB3C5180	219	219	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:32.340944 UTC	Disabled
0X8ADFCB3C0000	221	221	2	scsi_eh_14	0	0	0	0	2025-12-01 20:26:32.346113 UTC	Disabled
0X8ADFCAEF28C0	222	222	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:32.346305 UTC	Disabled
0X8ADFCAEF0000	223	223	2	scsi_eh_15	0	0	0	0	2025-12-01 20:26:32.346445 UTC	Disabled
0X8ADFCC1BD180	224	224	2	kworker/R-scsi_	0	0	0	0	2025-12-01	Disabled

									20:26:32.346869 UTC	
0X8ADFCC1B8000	225	225	2	scsi_eh_16	0	0	0	0	2025-12-01 20:26:32.347165 UTC	Disabled
0X8ADFCC1BA8C0	226	226	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:32.347250 UTC	Disabled
0X8ADFCC1C8000	227	227	2	scsi_eh_17	0	0	0	0	2025-12-01 20:26:32.347766 UTC	Disabled
0X8ADFCC0B5180	228	228	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:32.348061 UTC	Disabled
0X8ADFCC0B0000	229	229	2	scsi_eh_18	0	0	0	0	2025-12-01 20:26:32.3	Disabled

									48518 UTC	
0X8ADFCC27A8C0	234	234	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:32.350993 UTC	Disabled
0X8ADFCC10A8C0	235	235	2	scsi_eh_19	0	0	0	0	2025-12-01 20:26:32.352990 UTC	Disabled
0X8ADFCC10D180	236	236	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:32.356633 UTC	Disabled
0X8ADFCC108000	237	237	2	scsi_eh_20	0	0	0	0	2025-12-01 20:26:32.362799 UTC	Disabled
0X8ADFCC1CA8C0	238	238	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:32.363781 UTC	Disabled

0X8ADFCC1CD180	239	239	2	scsi_eh_21	0	0	0	0	2025-12-01 20:26:32.365186 UTC	Disabled
0X8ADFCC0B28C0	240	240	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:32.365785 UTC	Disabled
0X8ADFCC08D180	241	241	2	scsi_eh_22	0	0	0	0	2025-12-01 20:26:32.365961 UTC	Disabled
0X8ADFCC27D180	242	242	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:32.366121 UTC	Disabled
0X8ADFCAEF5180	243	243	2	scsi_eh_23	0	0	0	0	2025-12-01 20:26:32.366281 UTC	Disabled
0X8ADFCB3D28C0	244	244	2	kworker/R-scsi_	0	0	0	0	2025-12-01	Disabled

									20:26:32.366455 UTC	
0X8ADFCC0828C0	245	245	2	scsi_eh_24	0	0	0	0	2025-12-01 20:26:32.366818 UTC	Disabled
0X8ADFCC080000	246	246	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:32.367006 UTC	Disabled
0X8ADFCB3C8000	247	247	2	scsi_eh_25	0	0	0	0	2025-12-01 20:26:32.367473 UTC	Disabled
0X8ADFCB3C28C0	248	248	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:32.367595 UTC	Disabled
0X8ADFCC155180	249	249	2	scsi_eh_26	0	0	0	0	2025-12-01 20:26:32.3	Disabled

									67980 UTC	
0X8ADFCC150000	250	250	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:32.368117 UTC	Disabled
0X8ADFCC1528C0	251	251	2	scsi_eh_27	0	0	0	0	2025-12-01 20:26:32.368318 UTC	Disabled
0X8ADFCC2A8000	252	252	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:32.368473 UTC	Disabled
0X8ADFCC2AA8C0	253	253	2	scsi_eh_28	0	0	0	0	2025-12-01 20:26:32.368640 UTC	Disabled
0X8ADFCC2AD180	254	254	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:32.368713 UTC	Disabled

0X8ADFCC2C5180	255	255	2	scsi_eh_29	0	0	0	0	2025-12-01 20:26:32.368891 UTC	Disabled
0X8ADFCC2C0000	256	256	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:32.368955 UTC	Disabled
0X8ADFCC2C28C0	257	257	2	scsi_eh_30	0	0	0	0	2025-12-01 20:26:32.369138 UTC	Disabled
0X8ADFCC2D5180	258	258	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:32.369220 UTC	Disabled
0X8ADFCC2D0000	259	259	2	scsi_eh_31	0	0	0	0	2025-12-01 20:26:32.369410 UTC	Disabled
0X8ADFCC2D28C0	260	260	2	kworker/R-scsi_	0	0	0	0	2025-12-01	Disabled

									20:26:32.369570 UTC	
0X8ADFCC390000	279	279	2	kworker/u256:24	0	0	0	0	2025-12-01 20:26:32.376430 UTC	Disabled
0X8ADFCC1E5180	287	287	2	scsi_eh_32	0	0	0	0	2025-12-01 20:26:32.412188 UTC	Disabled
0X8ADFCC3AA8C0	288	288	2	kworker/R-scsi_	0	0	0	0	2025-12-01 20:26:32.412361 UTC	Disabled
0X8ADFCC020000	311	311	2	kworker/R-kdmfl	0	0	0	0	2025-12-01 20:26:32.792131 UTC	Disabled
0X8ADFCC038000	340	340	2	kworker/R-raid5	0	0	0	0	2025-12-01 20:26:34.1	Disabled

									35392 UTC	
0X8ADFCD398000	379	379	2	jbd2/dm-0-8	0	0	0	0	2025-12-01 20:26:34.444611 UTC	Disabled
0X8ADFCC03D180	380	380	2	kworker/R-ext4-	0	0	0	0	2025-12-01 20:26:34.444905 UTC	Disabled
0X8ADFCAFF5180	447	447	1	systemd-journal	0	0	0	0	2025-12-01 20:26:39.466557 UTC	Disabled
0X8ADFCD68A8C0	472	472	2	kworker/R-kmpat	0	0	0	0	2025-12-01 20:26:40.253285 UTC	Disabled
0X8ADFCD688000	473	473	2	kworker/R-kmpat	0	0	0	0	2025-12-01 20:26:40.253429 UTC	Disabled

0X8ADFC4F08000	474	474	1	multipathd	0	0	0	0	2025-12-01 20:26:40.254450 UTC	Disabled
0X8ADFC4D7A8C0	507	507	1	systemd-udevd	0	0	0	0	2025-12-01 20:26:40.825235 UTC	Disabled
0X8ADFC4CEA8C0	522	522	2	psimon	0	0	0	0	2025-12-01 20:26:42.682694 UTC	Disabled
0X8ADFC395A8C0	569	569	2	irq/57-vmw_vmci	0	0	0	0	2025-12-01 20:26:44.325434 UTC	Disabled
0X8ADFC39558000	570	570	2	irq/58-vmw_vmci	0	0	0	0	2025-12-01 20:26:44.327328 UTC	Disabled
0X8ADFC3A1A8C0	571	571	2	irq/59-vmw_vmci	0	0	0	0	2025-12-01	Disabled

									20:26:44.327512 UTC	
0X8ADFC3950000	576	576	2	irq/16-vmwgfx	0	0	0	0	2025-12-01 20:26:44.577421 UTC	Disabled
0X8ADFC3955180	577	577	2	kworker/R-ttm	0	0	0	0	2025-12-01 20:26:44.577633 UTC	Disabled
0X8ADFC4AC5180	691	691	2	jbd2/sda2-8	0	0	0	0	2025-12-01 20:26:46.663904 UTC	Disabled
0X8ADFC4AC0000	692	692	2	kworker/R-ext4-	0	0	0	0	2025-12-01 20:26:46.664185 UTC	Disabled
0X8ADFC4E5A8C0	881	881	1	systemd-network	998	998	998	998	2025-12-01 20:26:49.6	Disabled

									18361 UTC	
0X8ADFC74E8000	885	885	2	kworker/R-cfg80	0	0	0	0	2025-12-01 20:26:50.089163 UTC	Disabled
0X8ADFC4E5D180	903	903	1	systemd-resolve	992	992	992	992	2025-12-01 20:26:51.331260 UTC	Disabled
0X8ADFC74ED180	904	904	1	systemd-timesyn	997	997	997	997	2025-12-01 20:26:51.342693 UTC	Disabled
0X8ADFC4CE8000	906	906	1	VGAuthService	0	0	0	0	2025-12-01 20:26:51.348454 UTC	Disabled
0X8ADFC53D8000	907	907	1	vmtoolsd	0	0	0	0	2025-12-01 20:26:51.355783 UTC	Disabled

0X8ADFC4AC28C0	971	971	1	dbus-daemon	101	102	101	102	2025-12-01 20:26:52.594852 UTC	Disabled
0X8ADFCC4F5180	976	976	1	polkitd	991	991	991	991	2025-12-01 20:26:52.726511 UTC	Disabled
0X8ADFC7700000	981	981	1	systemd-logind	0	0	0	0	2025-12-01 20:26:52.761157 UTC	Disabled
0X8ADFC39528C0	982	982	1	udisksd	0	0	0	0	2025-12-01 20:26:52.766846 UTC	Disabled
0X8ADFC8ACA8C0	986	986	1	java	999	988	999	988	2025-12-01 20:26:53.010575 UTC	Disabled
0X8ADFC74F8000	990	990	1	java	110	110	110	110	2025-12-01	Disabled

									20:26:53.356674 UTC	
0X8ADFCC03A8C0	991	991	1	filebeat	0	0	0	0	2025-12-01 20:26:53.368481 UTC	Disabled
0X8ADFC8ACD180	992	992	1	node	111	111	111	111	2025-12-01 20:26:54.157734 UTC	Disabled
0X8ADFC91CA8C0	994	994	1	cron	0	0	0	0	2025-12-01 20:26:54.497575 UTC	Disabled
0X8ADFC91CD180	996	996	1	rsyslogd	103	104	103	104	2025-12-01 20:26:54.749483 UTC	Disabled
0X8ADFC8BC28C0	1007	1007	1	unattended-upgr	0	0	0	0	2025-12-01 20:26:54.8	Disabled

									50768 UTC	
0X8ADFC86DD180	1018	1018	1	login	0	1000	0	1000	2025-12-01 20:26:55.493210 UTC	Disabled
0X8ADFCD39A8C0	1294	1294	1	ModemManager	0	0	0	0	2025-12-01 20:27:01.206844 UTC	Disabled
0X8ADFC4838000	1423	1423	2	psimon	0	0	0	0	2025-12-01 20:27:14.924304 UTC	Disabled
0X8ADFCD68D180	1425	1425	1	systemd	1000	1000	1000	1000	2025-12-01 20:27:15.131972 UTC	Disabled
0X8ADFC483A8C0	1426	1426	1425	(sd-pam)	1000	1000	1000	1000	2025-12-01 20:27:15.519416 UTC	Disabled

0X8ADFCAC328C0	14 70	14 70	10 18	bash	10 00	10 00	10 00	100 0	2025-12-01 20:27:18.933962 UTC	Disabled
0X8ADFC4850000	16 12	16 12	99 0	java	11 0	11 0	110	110	2025-12-01 20:27:56.159561 UTC	Disabled
0X8ADFCA9B8000	16 39	16 39	16 12	controller	11 0	11 0	110	110	2025-12-01 20:28:21.633885 UTC	Disabled
0X8ADFC0DE8000	17 67	17 67	2	kworker/1:1	0	0	0	0	2025-12-01 20:33:03.036597 UTC	Disabled
0X8ADFC63628C0	17 97	17 97	1	fwupd	0	0	0	0	2025-12-01 20:39:29.537985 UTC	Disabled
0X8ADFC08C5180	18 06	18 06	1	upowerd	0	0	0	0	2025-12-01	Disabled

									20:39:46.857213 UTC	
0X8ADF60030000	1842	1842	2	kworker/u256:0	0	0	0	0	2025-12-01 20:46:40.163074 UTC	Disabled
0X8ADFC637D180	1854	1854	1	sshd	0	0	0	0	2025-12-01 20:48:43.573132 UTC	Disabled
0X8AE0008028C0	1940	1940	2	kworker/1:0H	0	0	0	0	2025-12-01 20:50:03.159634 UTC	Disabled
0X8AE0009A28C0	1948	1948	2	kworker/0:2	0	0	0	0	2025-12-01 20:50:16.089377 UTC	Disabled
0X8AE0009C0000	1986	1986	2	kworker/u256:1	0	0	0	0	2025-12-01 20:54:01.3	Disabled

									71271 UTC	
0X8ADFEAF30000	2079	2079	2	kworker/1:2H	0	0	0	0	2025-12-01 20:54:55.067567 UTC	Disabled
0X8ADF1F975180	2098	2098	2	kworker/1:2H	0	0	0	0	2025-12-01 20:58:26.525721 UTC	Disabled
0X8ADFC37A28C0	2185	2185	2	kworker/2:0H	0	0	0	0	2025-12-01 21:00:15.592528 UTC	Disabled
0X8ADF600078000	2202	2202	2	kworker/2:0H	0	0	0	0	2025-12-01 21:01:33.653695 UTC	Disabled
0X8ADED0028000	2215	2215	2	kworker/0:1H	0	0	0	0	2025-12-01 21:05:58.700135 UTC	Disabled

0X8ADED00F5180	22 20	22 20	2	kworker/u256:4	0	0	0	0	2025-12-01 21:09:28.656878 UTC	Disabled
0X8ADFC138A8C0	22 29	22 29	18 54	sshd	0	0	0	0	2025-12-01 21:11:09.714505 UTC	Disabled
0X8AE0009A0000	22 83	22 83	22 29	sshd	1000	1000	1000	1000	2025-12-01 21:11:15.164957 UTC	Disabled
0X8ADFCC395180	22 84	22 84	22 83	bash	1000	1000	1000	1000	2025-12-01 21:11:15.213969 UTC	Disabled
0X8ADFCC1E28C0	23 29	23 29	2	kworker/0:0	0	0	0	0	2025-12-01 21:14:19.739419 UTC	Disabled
0X8ADFC08C0000	23 30	23 30	2	kworker/2:1	0	0	0	0	2025-12-01	Disabled

									21:14:19.739578 UTC	
0X8ADFCC1E0000	2591	2591	1	packagekitd	0	0	0	0	2025-12-01 21:14:24.959222 UTC	Disabled
0X8AE000E2D180	2626	2626	2	kworker/0:0H	0	0	0	0	2025-12-01 21:14:40.455674 UTC	Disabled
0X8AE000800000	2640	2640	2	kworker/u256:2	0	0	0	0	2025-12-01 21:14:59.227275 UTC	Disabled
0X8AE000805180	2641	2641	2	kworker/u256:3	0	0	0	0	2025-12-01 21:14:59.437239 UTC	Disabled
0X8ADFC35ED180	3514	3514	2284	sudo	1000	0	0	0	2025-12-01 21:18:49.6	Disabled

									80883 UTC	
0X8ADFCC2FD180	35 15	35 15	35 14	sudo	10 00	0	0	0	2025-12-01 21:18:49.734187 UTC	Disabled
0X8ADF1F815180	35 16	35 16	35 15	insmod	0	0	0	0	2025-12-01 21:18:49.734640 UTC	Disabled
0X8ADF1F8128C0	35 18	35 18	2	kworker/1:1H	0	0	0	0	2025-12-01 21:18:52.173839 UTC	Disabled
0X8ADF1F810000	35 19	35 19	2	kworker/2:2H	0	0	0	0	2025-12-01 21:18:52.264076 UTC	Disabled
0X8ADFC422A8C0	35 24	35 24	2	kworker/0:2H	0	0	0	0	2025-12-01 21:20:07.841417 UTC	Disabled

0X8ADFC35E8000	35 22	35 22	2	kworker/u256:6	0	0	37	2E+09	2025-12-01 21:19:52.731265 UTC	Disabled
0X8AE0009C28C0	35 20	35 20	2	kworker/u256:5	0	0	37	2E+09	2025-12-01 21:19:22.488813 UTC	Disabled